

N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM
MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT
CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED
IN THE INTEREST OF MAKING AVAILABLE AS MUCH
INFORMATION AS POSSIBLE

TECHNICAL MEMORANDUM (NASA) 75

ANALYSIS OF FIRST AND SECOND ORDER BINARY QUANTIZED
DIGITAL PHASE-LOCKED LOOPS FOR IDEAL AND
WHITE GAUSSIAN NOISE INPUTS

Specific configurations of first and second order all digital phase-locked loops are analyzed for both ideal and additive white gaussian noise inputs. In addition, a design for a hardware digital phase-locked loop capable of either first or second order operation is presented along with appropriate experimental data obtained from testing of the hardware loop. All parameters chosen for the analysis and the design of the digital phase-locked loop are consistent with an application to an Omega navigation receiver although neither the analysis nor the design are limited to this application.

by

Paul R. Blasche

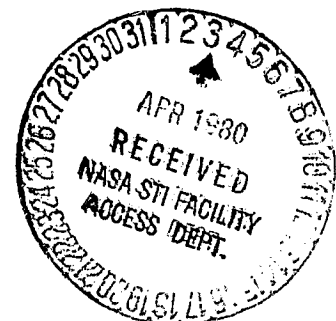
Avionics Engineering Center
Department of Electrical Engineering
Ohio University
Athens, Ohio 45701

March 1980

Supported by

National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia

Grant NGR 36-009-017



FOREWORD

This paper was submitted to and approved by the Department of Electrical Engineering and the College of Engineering and Technology of Ohio University, March 1980, as a dissertation.

TABLE OF CONTENTS

	PAGE
List of Figures	iii
I. INTRODUCTION	1
II. THE ANALOG PHASE-LOCKED LOOP	4
A. Introduction	4
B. The APLL with Ideal Input	5
C. The Linear APLL Model	6
III. THE DIGITAL PHASE-LOCKED LOOP	14
A. Introduction	14
B. The Digital Phase-Locked Loop	15
C. First Order DPLL Time Response	21
D. Second Order DPLL Time Response	25
IV. FIRST ORDER DPLL WITH ADDITIVE NOISE INPUT	38
A. Introduction	38
B. First-Order DPLL Configuration	38
C. Matrix of Transition Probabilities	44
D. Application of the Markov Chain Model	45
V. SECOND ORDER DPLL WITH ADDITIVE NOISE INPUT	57
A. Introduction	57
B. Second Order DPLL Configuration	57
C. Markov Chain Model of the Second Order DPLL	60
D. Application of the Markov Chain Model	67
VI. FIRST AND SECOND ORDER DPLL DESIGN	78
A. Introduction	78
B. DPLL Design and Testing	78
VII. CONCLUSIONS AND RECOMMENDATIONS	87
VIII. ACKNOWLEDGMENTS	90

	PAGE
IX. REFERENCES	91
X. APPENDICES	93
A. Markov Chains	94
B. Program Listings	99
1. PBSTGEN	99
2. PBDPL2	99
3. PBDPL2T	100
C. Binary Phase-Locked Loop Design	109
a. Binary Phase Detector	109
b. DIVIDE-by L Saturating Counter	109
c. Divide-by ($M \times N$) Counter	110
d. Variable Phase Reference Clock	110
e. Control Logic	111
D. Digital Low-Pass Filter	118

LIST OF FIGURES

	PAGE
Figure 2-1. Standard APLL Configuration.	4
Figure 2-2. Standard APLL Model.	7
Figure 2-3. Frequency Domain Model of Linear APLL.	8
Figure 2-4. Imperfect Integrator Loop Filter.	9
Figure 2-5. APLL Model with Fading Input.	13
Figure 3-1. Second-Order DPLL.	15
Figure 3-2. Waveform Sampling.	17
Figure 3-3. Response of First-Order DPLL to Phase Step Input.	23
Figure 3-4. Second Order DPLL Time Response.	25
Figure 3-5. Response of Second-Order DPLL to Phase Step Input.	27
Figure 3-6. Phase Error at First Zero Crossing for an Initial Phase Offset θ_1 and Sample Number k_1 of First Zero Crossing.	31
Figure 3-7. Value of First Peak Overshoot Versus the Phase at the First Zero Crossing.	31
Figure 3-8. Determination of Specified Response Parameters for an Initial Phase Offset.	32
Figure 3-9. Incremental Phase Plane Portrait for $\theta_1 = \pi/4$ and $\omega/\omega_o < 1$.	34
Figure 3-10. Incremental Phase Plane Portrait for $\theta_1 = \pi/4$ and $\omega/\omega_o > 1$.	35
Figure 4-1. First-Order DPLL Implementation.	49
Figure 4-2. First Order DPLL State Diagram.	50
Figure 4-3. Probability Density Function of Phase Error, $N=64$, $M=1$.	51
Figure 4-4. Probability Density Function of Phase Error, $N=64$, $SNR = 20$ dB.	52

	PAGE
Figure 4-5. Standard Deviation of Loop Phase Error Vs. Noise to Signal Ratio, $N = 64$.	53
Figure 4-6. Loop Transient Response $M = 1$, $N = 64$.	54
Figure 4-7. Loop Transient Response $M = 2$, $N = 64$.	55
Figure 4-8. Mean No. of Samples to Lock, $N = 64$, $M = 4$.	56
Figure 5-1. Second Order DPLL Implementation.	59
Figure 5-2. DPLL State Diagram, $N = 4$, $M = 2$, $L = 3$.	61
Figure 5-3. Steady State Phase Error $N = 64$, $M = 1$.	69
Figure 5-4. Steady State Phase Error $N = 64$, $M = 2$.	70
Figure 5-5. Steady State Phase Error $N = 64$, $M = 4$.	71
Figure 5-6. Loop Transient Response $M = 1$, $N = 64$, $L = 1$.	72
Figure 5-7. Loop Transient Response $M = 1$, $N = 64$, $L = 3$.	73
Figure 5-8. Loop Transient Response $M = 2$, $N = 64$, $L = 1$.	74
Figure 5-9. Loop Transient Response $M = 2$, $N = 64$, $L = 3$.	75
Figure 5-10. Loop Transient Response $M = 4$, $N = 64$, $L = 1$.	76
Figure 5-11. Loop Transient Response $M = 4$, $S = 3$, $N = 64$.	77
Figure 6-1. Hardware DPLL Block Diagram.	79
Figure 6-2. DPLL Test Circuit.	81
Figure 6-3. Probability Density Function for a Positive Sample at Phase Detector.	85
Figure 6-4. Probability Density Function for a Positive Sample at Phase Detector.	86
Figure C-1. DPLL Binary Phase Detector.	112
Figure C-2. S-Bit Saturating Counter.	113

	PAGE
Figure C-3. (M + N) Bit Counter with Adder Function.	114
Figure C-4. DPLL Variable Phase Reference Clock.	115
Figure C-5. Control Waveforms.	116
Figure C-6. Control Logic.	117
Figure D-1. Magnitude-Squared for Digital Low-Pass Filter.	119

CHAPTER I

INTRODUCTION

The phase-locked loop has long been recognized as a circuit with many important applications, and as such, the description of analog phase-locked loops (APLL's) has become well known as a large volume of material has been published to facilitate their use. In a survey of phase-locked loop development Gupta [11] has listed thirteen books and over 120 papers which discuss APLL design and applications. In recent years, there has been an increasing use of various types of loops employing discrete elements. Among these have been hybrid loops that utilize both analog and digital circuitry. Newer loop realizations have been circuits composed entirely of digital elements, the digital phase-locked loop (DPLL). The importance of these types of configurations lies in the relative ease of design and construction and, of equal importance, the ease in which such circuits can be maintained.

Unfortunately, the very attributes that make DPLL's attractive from a design and construction standpoint also contribute to difficulties in the theoretical analysis of DPLL operation and in this area the available literature is relatively thin. The first description of a general DPLL model was given by Reddy and Gupta [3] and further discussed by Gill and Gupta [10]. A more restricted but useful DPLL model for second order DPLL's has been presented by Weinburg and Liu [4]. However for both of these general DPLL models, little has been done to describe the response of the DPLL under specified ideal inputs.

In the area of fading input analysis of DPLL's the available literature may be separated into two approaches. The first approach as employed by Holmes and

Tegenelia [13], Weinburg and Liu [4], and Lee, Harington, and Cox [12], has been to model the DPLL configuration under study to operate linearly under some set of linearizing assumptions and then use classical techniques to determine the characteristics of the loop response. The second approach is more limited in that specific loop configurations are assumed and are then analyzed by random walk techniques. This method was first used by Holmes [6] and has the advantage that no linearizing assumptions are required. This approach has also been used by Cessna and Levy [5], Yamamoto and Mori [14], and Ransom and Gupta [15] for other specific loop configurations all of which employed resetting loop filters.

In the following chapters, both the time response and the steady state noise response of first- and second-order DPLL's will be discussed in detail. The loop configuration used throughout will be based on the model described by Weinburg and Liu [4], however the results will not be restricted by the linearizing assumptions made in reference [4]. Since the initial impetus for the study of the digital phase lock problem was provided by an application to an Omega navigation receiver, the practical realization of the DPLL's considered will be presented and all examples and parameter selection will be based on this application. However, neither the analysis techniques developed to describe the DPLL's operation nor the general hardware design presented are limited in any way to this application solely.

In Chapter II, the APLL is described to provide a background for the phase lock problem. In Chapter III, the response of the DPLL to specified ideal input signals is considered. In particular, new expressions are derived for frequency lock range

for first and second order DPLL's and a new partial solution for the difference equation describing second order DPLL operation is obtained. Chapters IV and V are devoted to the steady state noise analysis of first and second order DPLL's respectively. In both cases, the loops are modeled as first order Markov chains and from this model the steady state phase error and mean time to phase lock are determined for an input signal corrupted by white gaussian noise. For the first order DPLL, a loop employing a non-resetting loop filter is analyzed and found to have essentially equivalent steady state phase error as for the loops using Random-Walk loop filters described in references [5], [14], and [15], but with less circuit complexity. Chapter V presents a new analysis for a second order DPLL, using the Markov chain model, that is valid for all values of input signal-to-noise ratio. The previous methods of analysis for second order DPLL's as described in references [4] and [13] have involved linearizing assumptions that have limited the usefulness of the analysis to signal-to-noise ratios greater than 0.0 dB. Experimental verification of the Markov chain model was desired to show validity of the noise analysis of Chapters IV and V and therefore, in Chapter VI, a description of the design and construction of a first- and second-order DPLL is given. The experimental test data used in Chapters IV and V were obtained from this loop design.

CHAPTER II

THE ANALOG PHASE-LOCKED LOOP

A. Introduction. The standard analog phase-locked loop as shown in Figure 2-1 has been widely described in the literature for a variety of applications. The following brief description of the APLL is presented to develop the background material necessary for a comparison with the DPLL operational characteristics to be developed in the following chapters. In particular, it will be necessary to develop APLL response characteristics for both ideal and fading input signals.

As can be seen from Figure 2-1, the APLL consists only of a phase detector (multiplier), a linear filter, and a voltage-controlled oscillator (VCO) arranged in a feedback loop. In operation, the APLL is essentially a very narrowband filter whose characteristics are significantly dependent upon the type of linear filter employed in the loop. The derivation that follows is similar to that by Viterbi [1] and Lindsey and Simon [2].

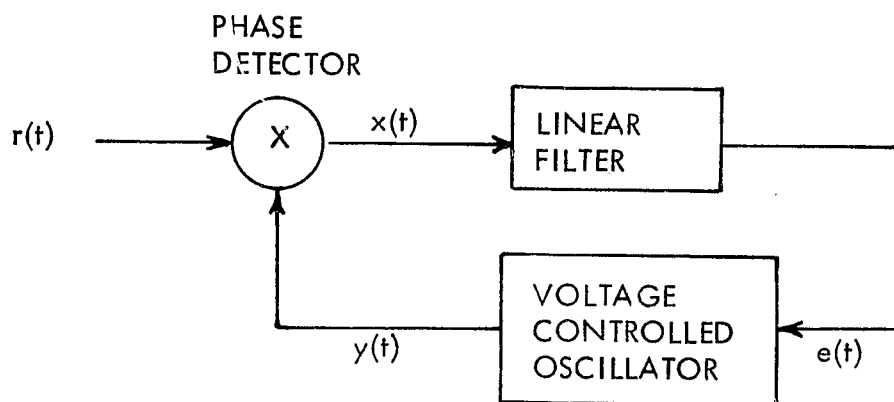


Figure 2-1. Standard APLL Configuration.

B. The APLL with Ideal Input. Consider the case of an ideal signal of the form,

$$r(t) = \sqrt{2} A \sin [\omega_o t + \theta(t)] \quad (2-1)$$

where A^2 is the total power of the input signal and ω_o is the quiescent frequency of the VCO. The input is angle modulated by $\theta(t)$ which is given by

$$\theta(t) = \Omega_o t + \theta_o(t) \quad (2-2)$$

where Ω_o is defined as the frequency offset from the VCO quiescent frequency and $\theta_o(t)$ is some function. The reference signal, $y(t)$ at the VCO output can be expressed as,

$$y(t) = \sqrt{2} K_1 \cos [\omega_o t + K_{VCO} \int e(\tau) d\tau] \quad (2-3a)$$

$$= \sqrt{2} K_1 \cos [\omega_o t + \hat{\theta}(t)] \quad (2-3b)$$

where K_1^2 is the total power of the reference signal, K_{VCO} is gain of the VCO with units rads/sec-volt, and

$$\hat{\theta}(t) = K_{VCO} \int e(\tau) d\tau \quad (2-4)$$

is the instantaneous phase estimate of the reference signal.

For the input and reference signals given, the error signal $x(t)$ at the phase detector output is given by,

$$x(t) = AK_1 K_m \sin [\theta(t) - \hat{\theta}(t)] \quad (2-5)$$

where the term involving $2\omega_o t$ has been ignored since it will be removed by the loop filter/VCO combination and K_m is the phase detector (multiplier) gain. If the initial conditions of the loop filter are zero, then the filter output $e(t)$ can be expressed as,

$$e(t) = A' K_m \int_0^t f(t - \tau) \sin [\theta(\tau) - \hat{\theta}(\tau)] d\tau \quad (2-6)$$

where $f(\tau)$ is the impulse response of the linear filter. Defining the phase error $\phi(t)$ for the loop as,

$$\phi(t) = \theta(t) - \hat{\theta}(t) \quad (2-7)$$

and substituting (2-6) into (2-4) and taking the derivative gives,

$$\frac{d\phi}{dt} = \frac{d\theta}{dt} - AK \int_0^t f(t - \tau) \sin \phi(\tau) d\tau \quad (2-8)$$

where $K = K_1 K_m K_{VCO}$ is defined as the loop gain. Thus given some input phase function $\theta(t)$, the solution of the integro-differential equation of (2-8) for $\phi(t)$ completely describes the operation of the APLL. Equation (2-8) also suggests the standard APLL model as given in Figure 2-2.

C. The Linear APLL Model. The APLL model of Figure 2-2 can be simplified and more readily described mathematically to a good approximation whenever the phase error is less than 0.5 radians. In this case the approximation

$$\sin \phi(t) \approx \phi(t) \quad (2-9)$$

can be made so that (2-8) becomes,

$$\frac{d\phi(t)}{dt} = \frac{d\theta(t)}{dt} - AK \int_0^t f(t - \tau) \phi(\tau) d\tau \quad (2-10)$$

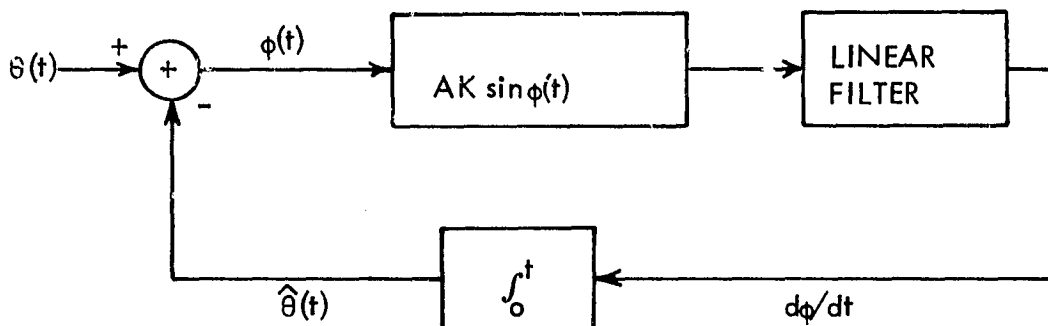


Figure 2-2. Standard APLL Model.

Taking the Laplace transform of (2-11) with initial conditions assumed to be zero gives the frequency domain phase error as,

$$\phi(s) = \frac{s}{s + AK F(s)} \theta(s) \quad (2-11)$$

Since $\phi(s) = \theta(s) - \hat{\theta}(s)$, equation (2-11) leads directly to,

$$\hat{\theta}(s) = \frac{AK F(s)}{s + AK F(s)} \theta(s) \quad (2-12a)$$

$$= H(s) \theta(s) \quad (2-12b)$$

where

$$H(s) = \frac{AK F(s)}{s + AK F(s)} \quad (2-13)$$

is the closed loop transfer function of the linear APLL. Equation (2-12a) suggests the s-domain linear APLL model shown in Figure 2-3.

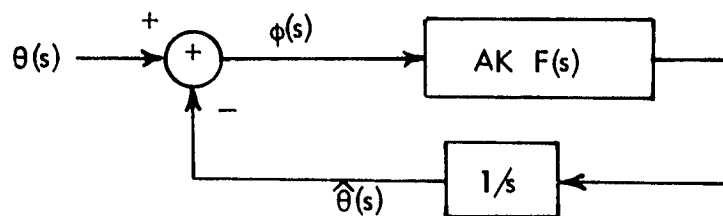


Figure 2-3. Frequency Domain Model of Linear APLL.

Under the linearizing assumption of (2-9), the APLL response as described by (2-11) and (2-12a) can be readily analyzed by classical techniques and definitions. Also, the APLL can be classified according to the characteristics of the open-loop transfer function $G(s)$,

$$G(s) = AK \frac{F(s)}{s} \quad (2-14)$$

In general for any control system, the order of the system is equal to the number of finite poles of the open-loop transfer function. Thus from (2-14) an APLL would be classified as a first-order system if there were no linear filter included in the loop. That is, $F(s) = 1$. Similarly, if,

$$F(s) = 1 + a/s \quad (2-15)$$

so that a signal-plus-integral loop filter is present in the APLL, then the APLL would be classified as a second-order system.

Using classical techniques, consider the case of a first-order APLL with a frequency step input. This corresponds to,

$$\theta(t) = \Omega_o t + \theta_o(t) \quad (2-16)$$

which after taking the Laplace transform with $\theta_o(t)$ a constant gives,

$$\theta(s) = \frac{\Omega_o}{s^2} + \frac{\theta_o}{s} \quad (2-17)$$

Substituting (2-17) into (2-11) with $F(s) = 1$ gives

$$\phi(s) = \frac{s}{s + AK} \left(\frac{\Omega_o}{s^2} + \frac{\theta_o}{s} \right) \quad (2-18)$$

From the final value theorem, the steady-state phase error for the APLL can be found as

$$\lim_{t \rightarrow \infty} \phi(t) = \Omega_o / AK \quad (2-19)$$

Thus the first-order APLL can achieve frequency synchronization but will not achieve perfect phase synchronization unless the frequency offset between the VCO and the input signal is zero.

Next consider the case in which the loop filter is the imperfect integrator of Figure 2-4. For this filter mechanization,

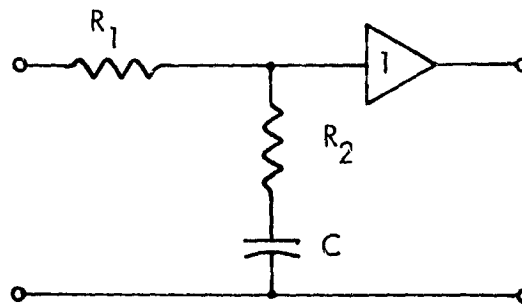


Figure 2-4. Imperfect Integrator Loop Filter.

$$F(s) = \frac{1 + \tau_2 s}{1 + \tau_1 s} \quad (2-20)$$

where

$$\tau_1 = (R_1 + R_2) C \quad (2-21)$$

$$\tau_2 = R_2 C \quad (2-22)$$

Substituting (2-20) and (2-17) into (2-11) gives,

$$\phi(s) = \frac{s(s + 1/\tau_1)}{s^2 + (AK\tau_2/\tau_1 + 1/\tau_1)s + \frac{AK}{\tau_1}} \left(\frac{\Omega_o}{s^2} + \frac{\theta_o}{s} \right) \quad (2-23)$$

and again applying the final value theorem gives the steady-state phase error as,

$$\lim_{t \rightarrow \infty} \phi(t) = \frac{\Omega_o}{AK/\tau_1} \quad (2-24)$$

which is equal to the steady-state phase error of the first-order loop attenuated by τ_1 . Since in practice it is usually desired that the filter be of the signal-plus-integral type, τ_1 will be a large number so that the pole of the loop filter is as near the origin as possible. Once again, this loop is capable of frequency synchronization but will track the phase with a constant offset.

D. The APLL in Additive Noise. The effects of additive noise on APLL operation may be studied by considering an input signal of the form,

$$r(t) = \sqrt{2} A \sin [\omega_o t + \theta(t)] + n_i(t) \quad (2-25)$$

where $n_i(t)$ is the narrowband noise process given by

$$n_i(t) = \sqrt{2} [x_c(t) \cos \omega_o t - x_s(t) \sin \omega_o t] \quad (2-26)$$

The terms $x_c(t)$ and $x_s(t)$ are assumed to be independent, stationary Gaussian white noise processes of zero mean and identical variances. Inclusion of the noise term gives a phase detector output of (referring to Figure 2-1)

$$x(t) = AK_1K_m \sin \phi(t) \quad (2-27)$$

$$-K_1K_m [X_c(t) \cos \phi(t) - X_s(t) \sin \phi(t)]$$

where

$$X_c(t) = x_c(t) \cos \theta(t) + x_s(t) \sin \theta(t) \quad (2-28a)$$

$$X_s(t) = x_c(t) \sin \theta(t) - x_s(t) \cos \theta(t) \quad (2-28b)$$

Note that again the terms involving $2\omega_o t$ have been ignored.

Following the same procedures as for the noiseless case, an integro-differential equation describing the APLL's operation is found to be,

$$\begin{aligned} d\phi/dt = d\theta/dt - K \int_0^t f(t-\tau) [A \sin \phi(\tau) \\ - X_c(\tau) \cos \phi(\tau) - X_s(\tau) \sin \phi(\tau)] d\tau \end{aligned} \quad (2-29)$$

As before, it is possible to represent the APLL by the simple model given in Figure 2-5.

Applying the linearizing assumption, $\sin \phi \approx \phi$ gives a simplified fading input model which can be analyzed by superposition. Thus it is only necessary that the noise terms be considered at this point since the effects of the deterministic portion of the signal was considered earlier. Considering just the noise term gives,

$$\phi(t) = \hat{\theta}(t) \quad (2-30)$$

which will have a noise spectral density given as,

$$S_{\phi}(\omega) = S_{\theta}(\omega) \quad (31a)$$

$$= \left| \frac{KF(\omega)/j\omega}{1 + AK^F(\omega)/j\omega} \right|^2 S_n(\omega) \quad (31b)$$

where $S_n(\omega)$ is the noise spectral density of the noise input. If the input noise is assumed to be white with single-sided density N_o then,

$$S_{\phi}(\omega) = \left| \frac{KF(\omega)/j\omega}{1 + AK^F(\omega)/j\omega} \right|^2 \frac{N_o}{2} \quad (2-32)$$

Recalling the closed-loop transfer function of (2-13), the phase error spectral density can be rewritten as,

$$S_{\phi}(\omega) = \frac{N_o}{2A^2} \left| H(j\omega) \right|^2 \quad (2-33)$$

The steady-state variance of the phase error then is easily found as,

$$\sigma_{\phi}^2 = \frac{N_o B_L}{A^2} \quad (2-34)$$

where B_L , the loop noise bandwidth, is defined as

$$B_L = \frac{1}{2\pi} \int_0^{\infty} \left| H(j\omega) \right|^2 d\omega \quad (2-35)$$

Thus once a loop filter has been specified, the loop noise bandwidth can be calculated and from this the operation of the APLL in the presence of noise is easily determined.

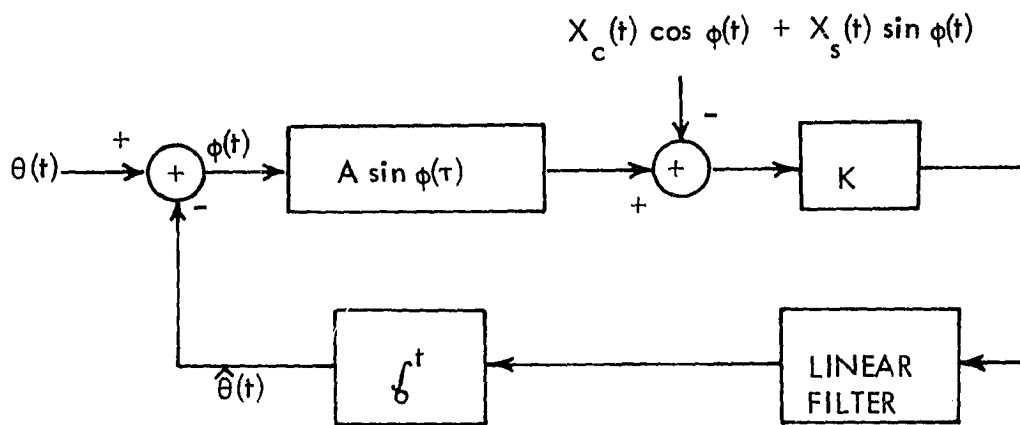


Figure 2-5. APLL Model with Fading Input.

CHAPTER III

THE DIGITAL PHASE-LOCKED LOOP

A. Introduction. For the APLL of Section II.B, it was possible to derive an integro-differential equation in terms of the input signal phase and the phase error of the tracking loop, the solution of which completely describes the APLL's operation. However, unlike the APLL a standard form of DPLL has not as yet developed in the literature. This is due in part to the non-linear operation of virtually all DPLL components. Still it is possible, for a specific DPLL configuration, to develop a difference equation that describes the DPLL's operation. This difference equation for the DPLL is analogous to the integro-differential equation of the APLL.

Difference equations have been derived for first and second-order DPLL's of a somewhat general configuration by Reddy and Gupta [3] and Wienburg and Liu [4]. However, the response characteristics of the DPLL's under consideration were largely investigated by iteration of the difference equations. This approach is basically a simulation of an ideal DPLL.

In the following, since the DPLL under consideration here is a slightly modified version of those given in references [3] and [4], the describing difference equations will be derived in detail. Further, it is very useful for a designer to have some knowledge of the response characteristics of a circuit without performing a simulation. Therefore, through analysis of the describing difference equation, the partial DPLL response characteristics for specified inputs are derived. This includes determination of time required to achieve phase lock for some initial phase offset and frequency range for phase acquisition.

B. The Digital Phase-Locked Loop. The DPLL configuration under consideration here is shown in Figure 3-1. For this loop the input is sampled at the positive-going zero crossing of the reference clock and the output of the phase detector is quantized to plus or minus one depending on the sign of the sample. In this manner, the phase difference measurement between the reference clock and the input signal is essentially reduced to a determination of phase lead or phase lag of the reference clock with respect to the input signal. The two forward paths represent a filter in the sense that they alter the phase detector output to create a signal suitable for adjustment of the reference clock phase. Notice that the reference clock phase will take on a finite number of values determined by Δ_1 and Δ_2 .

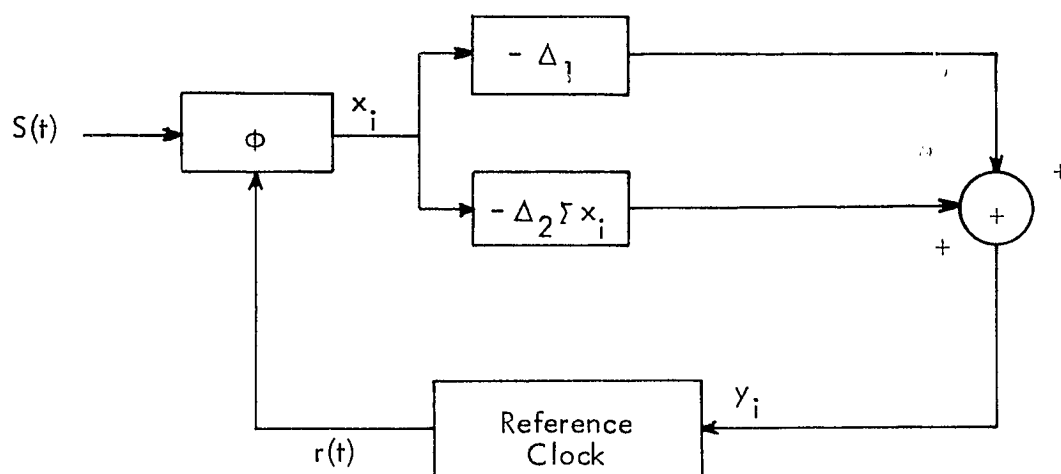


Figure 3-1. Second-Order DPLL.

Operation of the loop can best be described by considering the case of an ideal input signal of the form,

$$s(t) = A_c \sin [\omega_o t + \theta(t)] \quad (3-1)$$

where

$$\theta(t) = (\omega_o - \omega) t + \theta_i(t) \quad (3-2)$$

defines a frequency offset plus a modulating phase function. The reference clock is given by,

$$r(t) = \sin [\omega_o t + \theta_o(t)] \quad (3-3)$$

which is assumed to be quantized to N distinct phase states so that (3-3) may be written as,

$$r(t) = \sin [\omega_o t + \frac{\pi}{N} (N - 2i + 1)] \quad i = 1, 2, \dots, N \quad (3-4)$$

For example if the phase state of the reference clock is

$$i = N/4 \quad (3-5)$$

when the reference clock is given by

$$r(t) = \sin [\omega_o t - \frac{\pi}{N} (\frac{N-2}{2})] \quad (3-6)$$

For the sampling example depicted by Figure 3-2, the input is sampled at the positive-going zero crossing of the reference clock giving a positive value for the first sample. Since this first sample is positive in sign, the output of the phase detector, $X(1)$ is +1. This phase detector output is then modified by the two forward paths to produce the signal,

$$y(1) = -\Delta_1 - \Delta_2 \text{ radians} \quad (3-7)$$

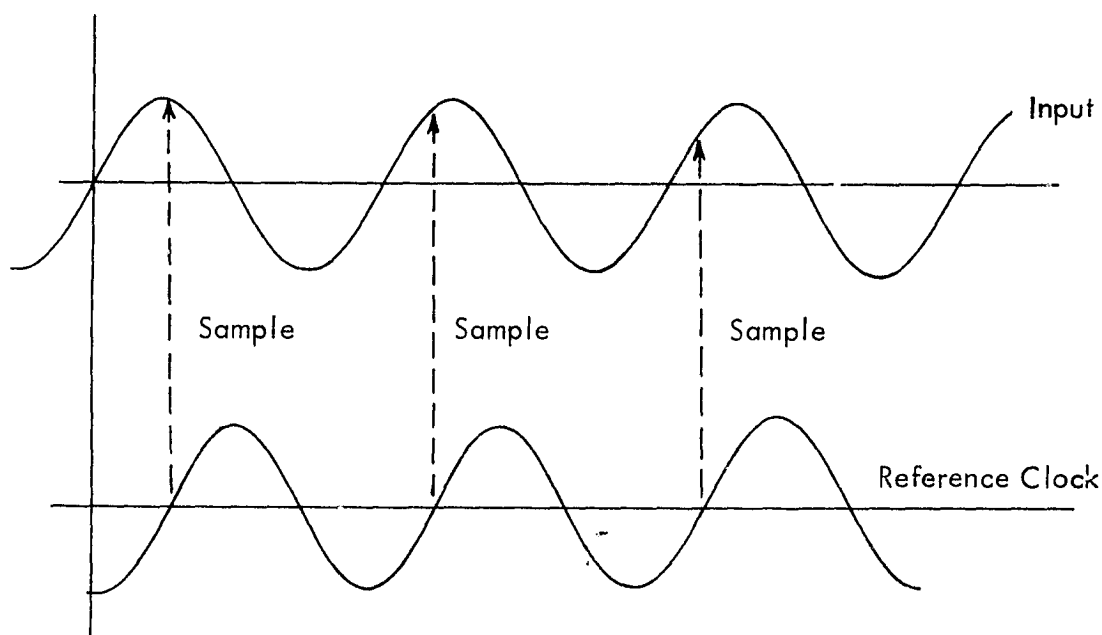


Figure 3-2. Waveform Sampling.

which is the phase increment by which the reference clock is altered. Obviously, the values of Δ_1 and Δ_2 are related to the quantization of the loop. If at the second sample of the input signal (the second positive-going zero crossing of the reference clock) the reference clock still lags the input, then the reference clock will be altered in phase by,

$$y(2) = -\Delta_1 - 2\Delta_2 \text{ radians} \quad (3-8)$$

Similarly, if the lag condition still exists at the third sample, then the reference clock will be altered in phase by,

$$y(3) = -\Delta_1 - 3\Delta_2 \text{ radians} \quad (3-9)$$

This condition will continue until the j -th sample, at which time the reference clock phase leads the input signal phase and the reference clock will be updated in phase by,

$$y(i) = + \Delta_1 - \Delta_2 \sum_{i=1}^i X(i) \quad (3-10a)$$

$$= + \Delta_1 - (i-1)\Delta_2 \quad (3-10b)$$

Similarly, if at the $(i+1)$ -th sample the reference clock still leads the input then,

$$y(i+1) = + \Delta_1 + 2\Delta_2 - \Delta_2 (i-1) \quad (3-11)$$

The loop updating will continue in this manner until the DPLL achieves a lock condition which is characterized by

$$y(i) \leq \left| \Delta_1 + \Delta_2 \right| + \left| (\omega_o - \omega) T(i) \right| \quad (3-12)$$

for all successive samples with $T(i)$ defined as the time interval between the i^{th} and $(i-1)$ -th sample.

For the analog phase-locked loop described in Section II.B, it was possible to study loop characteristics from the solution of an integro-differential equation that modeled the loop's behavior. In an analogous manner, it is possible to develop a difference equation for the digital phase-locked loop that characterizes the loop phase error on a sample-to-sample basis. Because the loop phase error is determined in terms of loop sample number, it is also necessary to develop an equation describing the time of occurrence of a particular loop sample. The combined use of these two discrete equations then will completely model the operation of the DPLL for an arbitrary phase input. The remainder of this section then will develop in detail these two describing equations.

Proceeding in a manner similar to [3], the total phase of the input signal given by (3-1) is,

$$\beta_i(t) = \omega_o t + \theta(t) \quad (3-13)$$

and the total phase of the reference clock given by (3-3) is,

$$\beta_o(t) = \omega_o t + \theta_o(t) \quad (3-14)$$

However, the input signal can be sampled only at discrete times and the reference clock phase can exist only in specified states so that (3-13) and (3-14) are more properly written as,

$$\beta_i(k) = \omega_o t(k) + \theta(k) \quad (3-15)$$

and

$$\beta_o(k) = \omega_o t(k) + \theta_o(k) \quad (3-16)$$

where for simplicity $f[t(k)]$ is written as $f(k)$ with k representing the sample number of the loop.

The output of the phase detector is a sequence of values ± 1 , the sign being determined by whether the phase of the reference clock leads or lags the phase of the input signal. Thus the phase detector output sequence, $X(k)$, may be written as,

$$X(k) = \text{sgn} [\sin [\beta_i(k) - \beta_o(k)]] \quad (3-17a)$$

$$= \text{sgn} [\sin [\phi(k)]] \quad (3-17b)$$

where

$$\phi(k) = \theta(k) - \theta_o(k) \quad (3-18)$$

is defined as the phase error for the DPLL. Further, the phase of the reference clock is altered by a value $Y(k)$ radians at the k -th sample so that the phase of the reference clock is given as,

$$\theta_o(k) = \sum_{i=1}^{k-1} Y(i) + \theta_o(o) \quad (3-19)$$

where $\theta(0)$ is the initial phase of the reference clock which will be assumed to be zero. Substituting (3-2) and (3-19) into (3-18) gives the loop phase error as,

$$\phi(k) = (\omega_o - \omega) t(k) + \theta_i(k) - \sum_{i=1}^{k-1} Y(i) \quad (3-20)$$

As was stated previously, it is desired to develop a difference equation that is independent of time. Therefore, it is necessary to determine an expression for $t(k)$ in (3-20) that is dependent only on the sample number k . First, define the value $T(k)$ as the time interval between samples so that,

$$T(k) = t(k) - t(k-1) \quad (3-21)$$

However, this can also be expressed as,

$$T(k) = T - \frac{T}{2\pi} Y(k-1) \quad (3-22)$$

where $T = \frac{2\pi}{\omega_o}$ is the period of the quiescent reference clock. The time of occurrence of the k -th sample then is just the sum of all $T(i)$, $i \leq k$, or,

$$t(k) = \sum_{i=1}^k T(k) \quad (3-23a)$$

$$= kT - \frac{T}{2\pi} \sum_{i=1}^{k-1} Y(i) \quad (3-23b)$$

Substituting (3-23b) into (3-20) gives an expression for phase error, after algebraic manipulation, as,

$$\phi(k) = \theta_i(k) + \frac{\omega_o - \omega}{\omega_o} 2\pi k - \frac{\omega}{\omega_o} \sum_{i=1}^{k-1} Y(i) \quad (3-24)$$

and the sample-to-sample difference in phase error is,

$$\phi(k+1) - \phi(k) = \theta_i(k+1) - \theta_i(k) + 2\pi \frac{\omega_o - \omega}{\omega_o} - \frac{\omega}{\omega_o} Y(k) \quad (3-25)$$

The value $Y(k)$ is defined as the phase increment of the reference clock for the k -th sample and will take on values determined by the gain constants Δ_1 and Δ_2 of,

$$Y(k) = \Delta_1 X(k) + \Delta_2 \sum_{i=1}^k X(i) \quad (3-26a)$$

$$= \Delta_1 \operatorname{sgn} [\sin \phi(k)] + \Delta_2 \sum_{i=1}^k \operatorname{sgn} [\sin \phi(i)] \quad (3-26b)$$

Substituting (3-26b) into (3-25) gives the phase error difference equation for the DPLL as,

$$\phi(k+1) - \phi(k) = \theta_i(k+1) - \theta_i(k) + 2\pi \frac{\omega_o - \omega}{\omega_o} \quad (3-27)$$

$$- \frac{\omega}{\omega_o} \left\{ \Delta_1 \operatorname{sgn} [\sin \phi(k)] + \Delta_2 \sum_{i=1}^k \operatorname{sgn} [\sin \phi(i)] \right\}$$

To be complete, it is necessary to express (3-23b) in terms of the phase error by substituting (3-26b) into (3-23b) giving,

$$t(k) = kT - \frac{T}{2\pi} \sum_{i=1}^{k-1} \left\{ \Delta_1 \operatorname{sgn} [\sin \phi(i)] + \Delta_2 \sum_{j=1}^i \operatorname{sgn} [\sin \phi(j)] \right\} \quad (3-28)$$

The DPLL then is completely described by the equation pair, (3-27) and (3-28), the former describing the phase error of the loop and the latter describing the discrete time at which the phase of the reference clock may change state.

C. First Order DPLL Time Response. A first order DPLL results for the system of Figure 3-1 when the gain term Δ_2 is set to zero. For this case (3-27) and (3-28) become, respectively,

$$\begin{aligned} \phi(k+1) - \phi(k) &= \theta_i(k+1) - \theta_i(k) + 2\pi \frac{\omega_o - \omega}{\omega_o} \\ &= \frac{\omega}{\omega_o} \Delta_1 \operatorname{sgn} \phi(k) \end{aligned} \quad (3-29)$$

and

$$\phi(k) = kT - \frac{T}{2\pi} \Delta_1 \sum_{i=1}^{k-1} \operatorname{sgn} \phi(i) \quad (3-30)$$

Note that for first order DPLL response the phase error is constrained to

$$|\phi(k)| \leq \pi \quad (3-31)$$

so that

$$\operatorname{sgn} [\sin \phi(k)] = \operatorname{sgn} \phi(k) \quad (3-32)$$

As defined previously, the total number of phase states of the reference clock is N .

Since at each sample of the input signal the reference clock phase is incremented by a value $\pm \Delta_1$, then

$$N = 2\pi / \Delta_1 \quad (3-33)$$

so that (3-30) can be written as,

$$\phi(k) = kT - \frac{T}{N} \sum_{i=1}^{k-1} \operatorname{sgn} \phi(i) \quad (3-34)$$

Consider first the response of the first order DPLL to an input signal whose frequency is identical to that of the DPLL's reference clock but differs initially in phase by some constant θ_i . For this case, $\omega_o = \omega$ and the input phase will be a constant with $\theta_i(k+1) = \theta_i(k)$ for all k which without loss of generality can be assumed to be zero. Thus, (3-29) becomes

$$\phi(k+1) - \phi(k) = -\Delta_1 \operatorname{sgn} \phi(k) \quad (3-35)$$

with the boundary condition $\phi(0) = \theta_i$. For this simple case, the loop will be up-

dated by a value Δ_1 radians at each sample until the phase error changes sign at which point the quantized phase error will oscillate about the true phase of the input. An example response is given in Figure 3-3. It is evident that a lock condition occurs whenever

$$|\phi(k)| \leq \Delta_1 \quad \text{for all } k \geq K \quad (3-36)$$

where K is given as,

$$K = \text{Integer} \lceil \theta_1 / \Delta_1 \rceil - 1 \quad (3-37)$$

The time of occurrence of the lock condition can then be found by evaluating (3-34) for K given by (3-37). Note that for all $k \leq K$ the value of $\text{sgn } \phi(k)$ will not change.

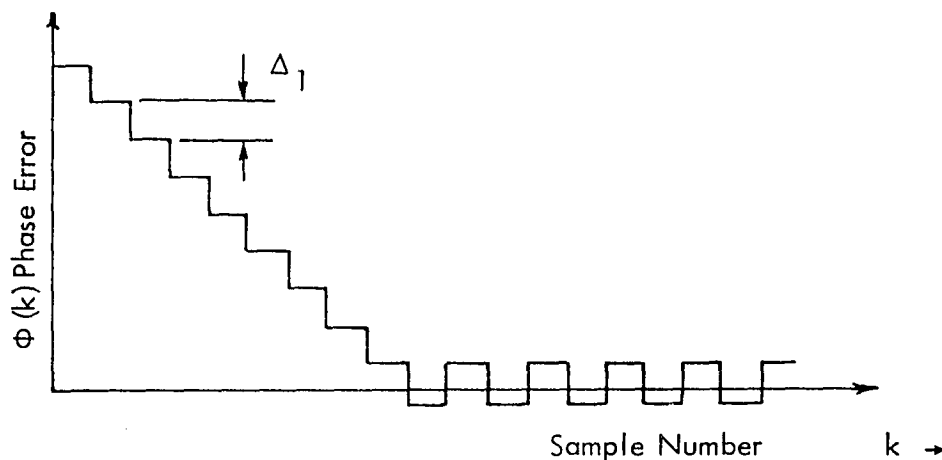


Figure 3-3. Response of First-Order DPLL to Phase Step Input.

Next consider the case where the input signal is offset by a constant frequency from the first order DPLL reference clock. For this condition the phase error difference equation of (3-27) is given as,

$$\phi(k+1) - \phi(k) = 2\pi \frac{\omega_o - \omega}{\omega_o} - \frac{\omega}{\omega_o} \Delta_1 \operatorname{sgn} \phi(k) \quad (3-38)$$

where it has also been assumed that $\theta_i(k)$ is equal to zero. In reference [3] an expression was obtained for frequency lock range by making the approximation $\phi(k) \approx 0$ in the steady state which ignores the quantization of the loop. However, it is possible to include the effects of quantization in determining frequency lock range. Recalling the condition for first order DPLL lock as given by (3-36), then an equivalent condition for lock is,

$$|\phi(k+1) - \phi(k)| \leq 2\Delta_1 \quad (3-39)$$

which after some algebraic manipulation gives

$$\frac{N-2}{N-1} \leq \frac{\omega}{\omega_o} \leq \frac{N+2}{N+1} \quad (3-40)$$

This expression is plotted in Figure 3-4 and gives the normalized frequency range over which the first order DPLL can remain locked as a function of the number of states of the DPLL reference clock. Note that the undefined lower limit for $N=1$ is of no consequence since this would correspond to a DPLL with only one phase state.

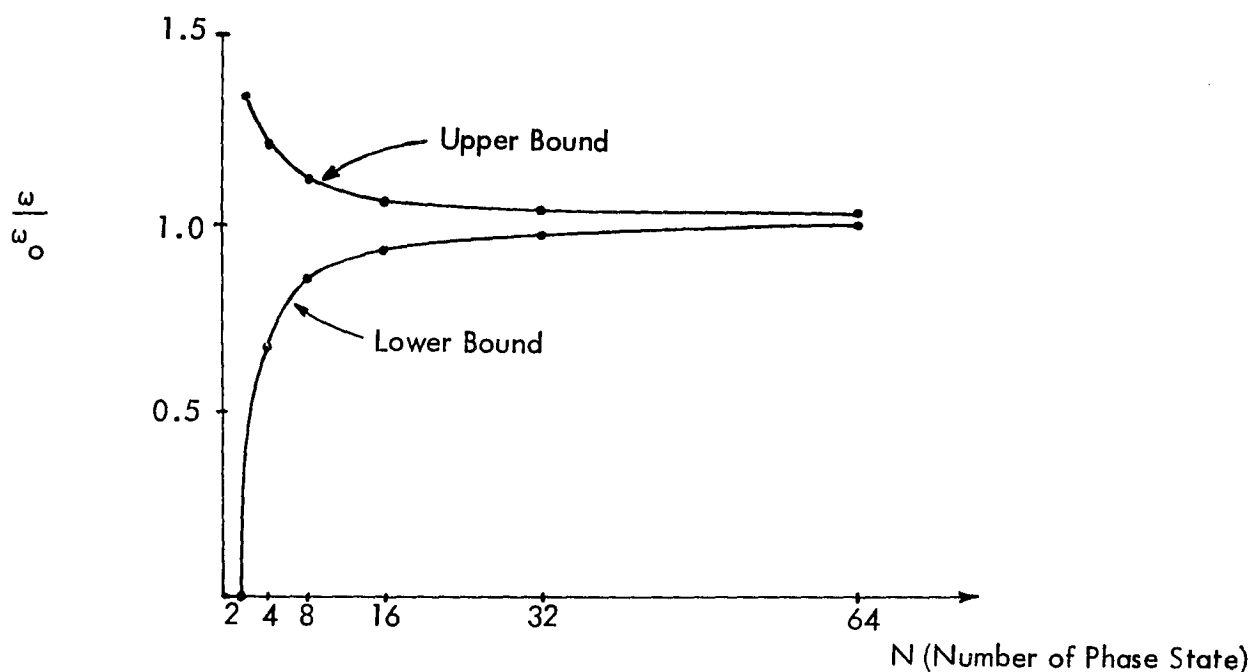


Figure 3-4. First Order DPLL Lock Bounds for Frequency Offset.

D. Second Order DPLL Time Response. The time response for a first order DPLL has been characterized in the previous section for specified inputs. Notice however that the characterization of the time response did not involve the solution of the phase error difference equation because of the nonlinear nature of the equation. Likewise, it is not possible to find a direct solution for the phase error difference equation for a second order DPLL. Furthermore, no significant discussion of second order DPLL response has been found in the literature for the generalized model of Figure 3-1. The following, while not giving a complete solution, fills some of this void by determining valid and pertinent characteristics of the second order DPLL time response for specified input conditions.

As with the first order DPLL, the first case to be considered for the second order loop is an input signal with a constant initial phase offset θ_1 and zero frequency difference with respect to the reference clock. Also, the input signal phase will again be assumed to be zero radians and the constraint of (3-31) is applicable so that (3-27) becomes

$$\phi(k+1) - \phi(k) = -\Delta_1 \operatorname{sgn} \phi(k) - \Delta_2 \sum_{i=1}^k \operatorname{sgn} \phi(i) \quad (3-41)$$

with $\phi(0) = \theta_1$. Since the total number of phase states of the reference clock is N , it is necessary that some relationship exist between Δ_1 , Δ_2 , and N . For the second order loop it is assumed that Δ_1 and Δ_2 are integrally related as,

$$\Delta_1 = n \Delta_2 \quad (3-42)$$

and the total number of phase states will be given as,

$$N = 2\pi/\Delta_1 = 2\pi/n \Delta_2 \quad (3-43)$$

The typical phase error response versus the loop sample number as given by (3-41) is shown in Figure 3-5. While a complete solution of (3-41) is not performed it would be of interest to determine the sample number of each zero crossing of the phase error function and both the sample number and the value of each peak overshoot of the phase error function up to the occurrence of phase lock. A knowledge of these values not only characterizes pertinent parts of the phase error function but can be used in conjunction with one another to give the total time required to achieve phase lock for an initial phase offset. Therefore, it will be necessary to determine the values $k_1, k_2, \dots, k_i, k_{p1}, k_{p2}, \dots, k_{pi}$, and

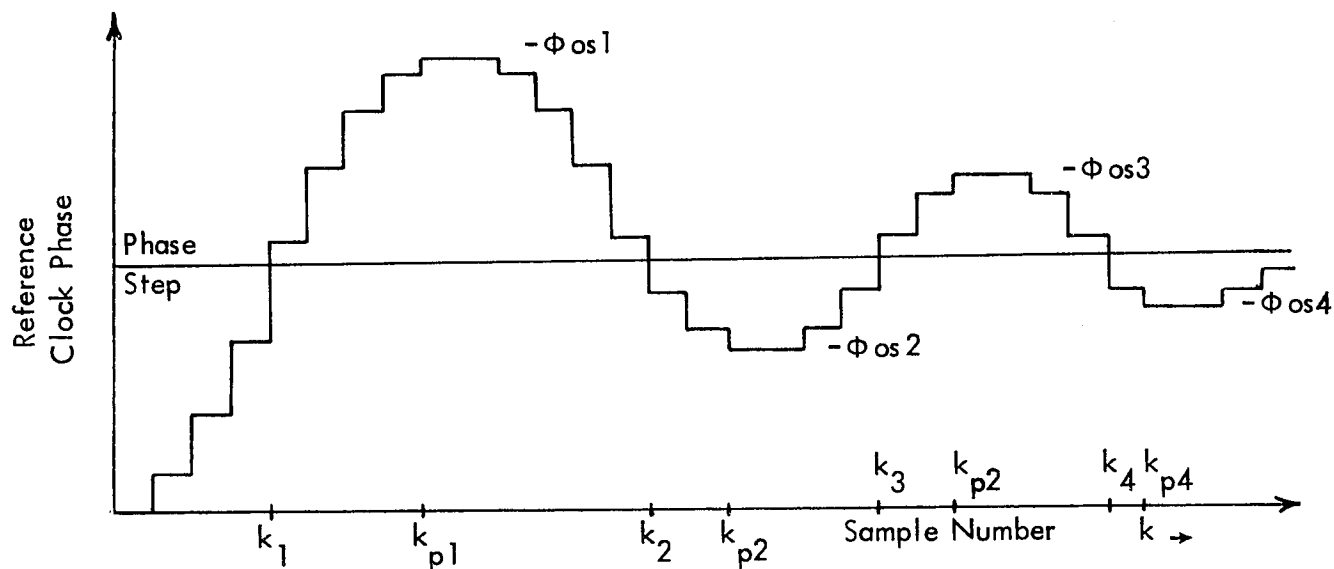


Figure 3-5. Response of Second-Order DPLL to Phase Step Input.

$\phi_{os1}, \phi_{os2}, \dots, \phi_{osi}$ up to the value,

$$\phi_{osi} \leq |\Delta_1 + \Delta_2| \quad (3-44)$$

at which time a lock condition has been achieved at k_i .

If the initial phase offset is assumed to be $\theta_1 \geq 0$, then the first zero crossing of the phase error function occurs when

$$\phi(k_1) \leq 0 \quad (3-45)$$

This value of k_1 may be found by considering the phase error for values of $k \leq k_1$

which is given as,

$$\phi(k) = \theta_1 - k \Delta_1 - \Delta_2 \sum_{i=1}^k i \quad (3-46)$$

The first zero crossing of the phase error function occurs at

$$k \Delta_1 + \Delta_2 \sum_{i=1}^k i \geq \theta_1 \quad (3-47)$$

However,

$$\sum_{i=1}^k i = \frac{k(k+1)}{2} \quad (3-48)$$

which may be substituted into (3-47) to give (after rearranging terms),

$$k^2 + \frac{2\Delta_1 + \Delta_2}{\Delta_2} k - \frac{2}{\Delta_2} \theta_1 \geq 0 \quad (3-49)$$

The positive root solution for (3-49) evaluated for equality to zero gives the number

of samples required for the first zero crossing of the phase error function as,

$$k_1 = \frac{-(2\Delta_1 + \Delta_2)}{2\Delta_2} + \left[\left(\frac{2\Delta_1 + \Delta_2}{2\Delta_2} \right)^2 + \frac{\theta_1}{\Delta_2} \right]^{\frac{1}{2}} \quad (3-50)$$

After the k_1 -th sample the sign of the phase detector output changes giving the phase error as,

$$\phi(k_1 + 1) = \phi(k_1) + \Delta_1 - \Delta_2 (k_1 - 1) \quad (3-51)$$

which can be generalized to

$$\phi(k_1 + i) = \phi(k_1) + i\Delta_1 - \frac{\Delta_2}{2} i(2k_1 - i - 1) \quad (3-52)$$

for $k_1 + i \leq k_{pi}$. Defining k_{pi} as the sample for which the peak overshoot occurs following the k_1 zero crossing, then it is obvious that,

$$\phi(k_{pi}) \leq \phi(k_{pi} + 1) \quad (3-53)$$

Applying the constraint of (3-53) to (3-52) gives the sample at which peak overshoot occurs as,

$$k_{pi} = 2k_1 - \Delta_1 / \Delta_2 - 1 \quad (3-54)$$

$$= 2k_1 - n - 1 \quad (3-55)$$

The value of the peak overshoot is found by evaluating (3-52) at $i = k_{p1} + k_1$ with k_{p1} determined from (3-55) giving,

$$\phi(k_{p1}) = \phi(k_1) + (k_1 - n - 1) \left[\Delta_1 - \frac{\Delta_2}{2} (k_1 + n) \right] \quad (3-56)$$

It is important to note that at the peak overshoot point that the value contained in the summation block of Figure 3-1 will be zero.

The partial response of the loop giving the zero crossings and the peak overshoot values for the phase error function can be found by repeated application of (3-50) and (3-56). Notice that since the summation term of the phase error function is zero at sample k_{pi} , then k_{i+1} can be found from (3-50) where the initial phase offset is given by ϕ_{osi} . Further, once the values of Δ_1 and Δ_2 have been specified it is a simple manner to determine the required response values graphically. As an example, consider the case for which $\Delta_1 = \Delta_2 = \frac{2\pi}{32}$. For this case (3-50) reduces to,

$$k_1 = -1.5 + (2.25 + 10.2 \theta_1)^{\frac{1}{2}} \quad (3-57)$$

and the phase error of the first zero crossing is determined from (3-46) as,

$$\phi(k) = \theta_1 - 0.196 (k_1^2 + 3k_1) \quad (3-58)$$

Equation (3-58) is in slope-intercept form with the intercept determined by (3-57).

Notice however that the value of k_1 determined by (3-57) applies only over a certain range of θ_1 and therefore (3-58) is likewise applicable only over the same

range of θ_1 . Thus, when (3-58) is plotted in Figure 3-6 it takes on a saw-tooth characteristic. Substituting the loop gain values into (3-56) gives the value of the first peak overshoot as,

$$\phi(k_{p1}) = \phi(k_1) + 0.098 (k_1 - 2) (1 - k_p) \quad (3-59)$$

which is also in slope-intercept form and is plotted in Figure 3-7. Notice that when the zero crossing occurs at the second loop sample, the loop will be in phase lock.

It is noted that the phase error at the occurrence of the first zero crossing of the phase error is plotted on the ordinate of Figure 3-6 and on the abscissa of Figure 3-7. Further, since the i -th zero crossing parameters are found by using the above procedure with the initial phase offset equal to the value of the $(i-1)$ -th peak overshoot, then the abscissa variable of Figure 3-6 is equal to the ordinate variable of Figure 3-7. Therefore it is possible to combine the results of these two plots. This is done in Figure 3-8 where (3-58) for $\theta_1 \geq 0$ is plotted in the first quadrant and for $\theta_1 \leq 0$ in the third quadrant. Similarly, (3-59) for a negative phase error at the zero crossing is plotted in the second quadrant and for a positive phase error at the zero crossing in the fourth quadrant.

Using Figure 3-8 it is possible to determine the phase error characteristics during loop phase acquisition as follows. Assume that the initial phase offset of the loop is $\theta_1 = 2.75$ radians as shown in Figure 3-8. The first zero crossing of the phase error function will occur at the fifth sample and the phase error will be -0.98 radians. The value of the first peak overshoot is the projection on the $-\theta_1$ axis of the intersection of $\phi(k) = 0.98$ radians with the $k=5$ line in the second quadrant which is equal to -2.04 radians. This value of first peak overshoot is then used

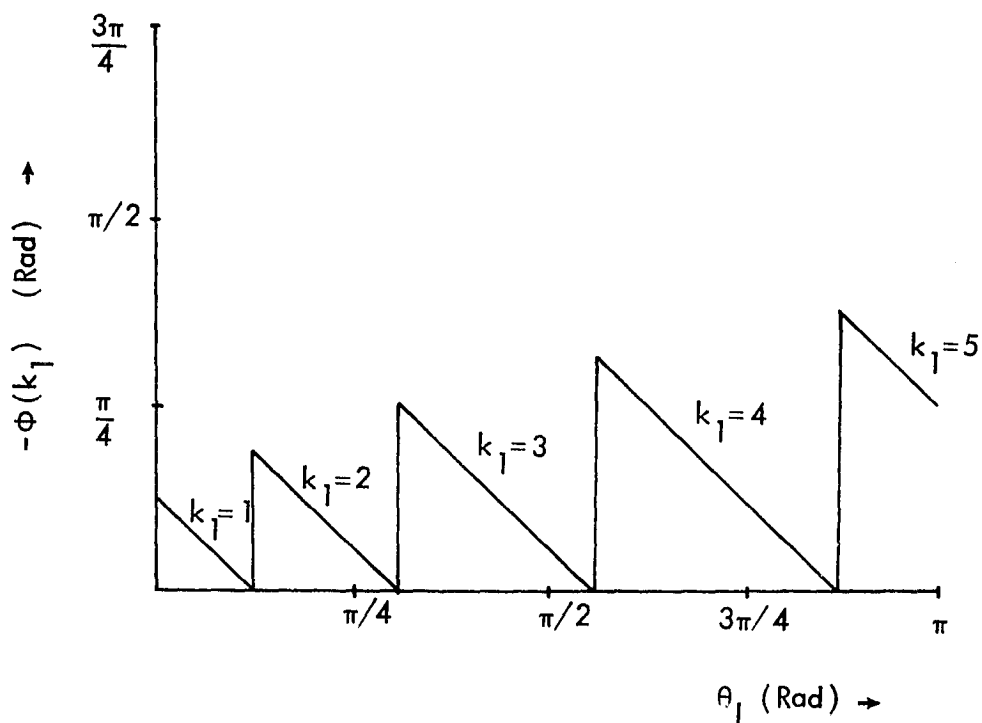


Figure 3-6. Phase Error at First Zero Crossing for an Initial Phase Offset θ_1 and Sample Number k_1 of First Zero Crossing.

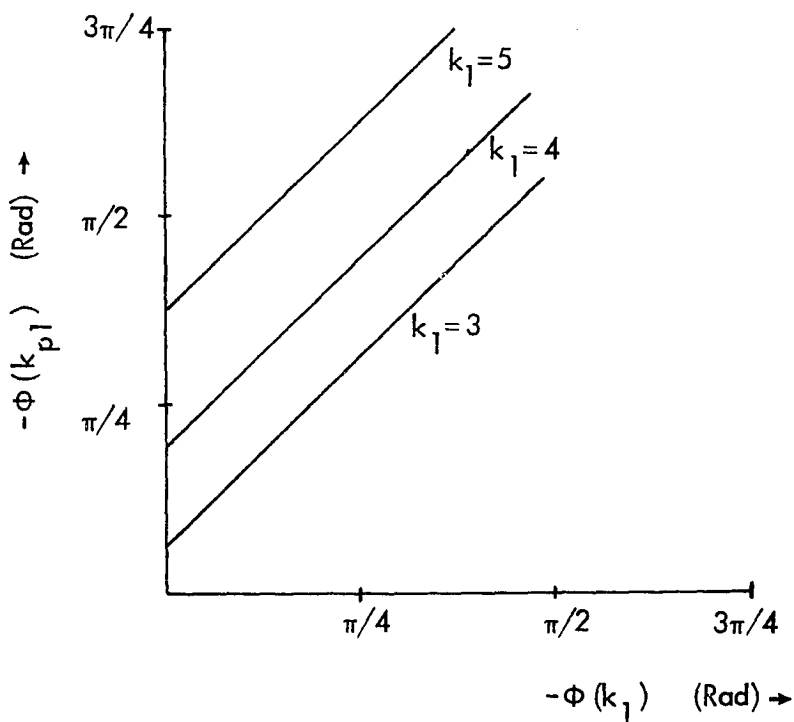


Figure 3-7. Value of First Peak Overshoot Versus the Phase at the First Zero Crossing.

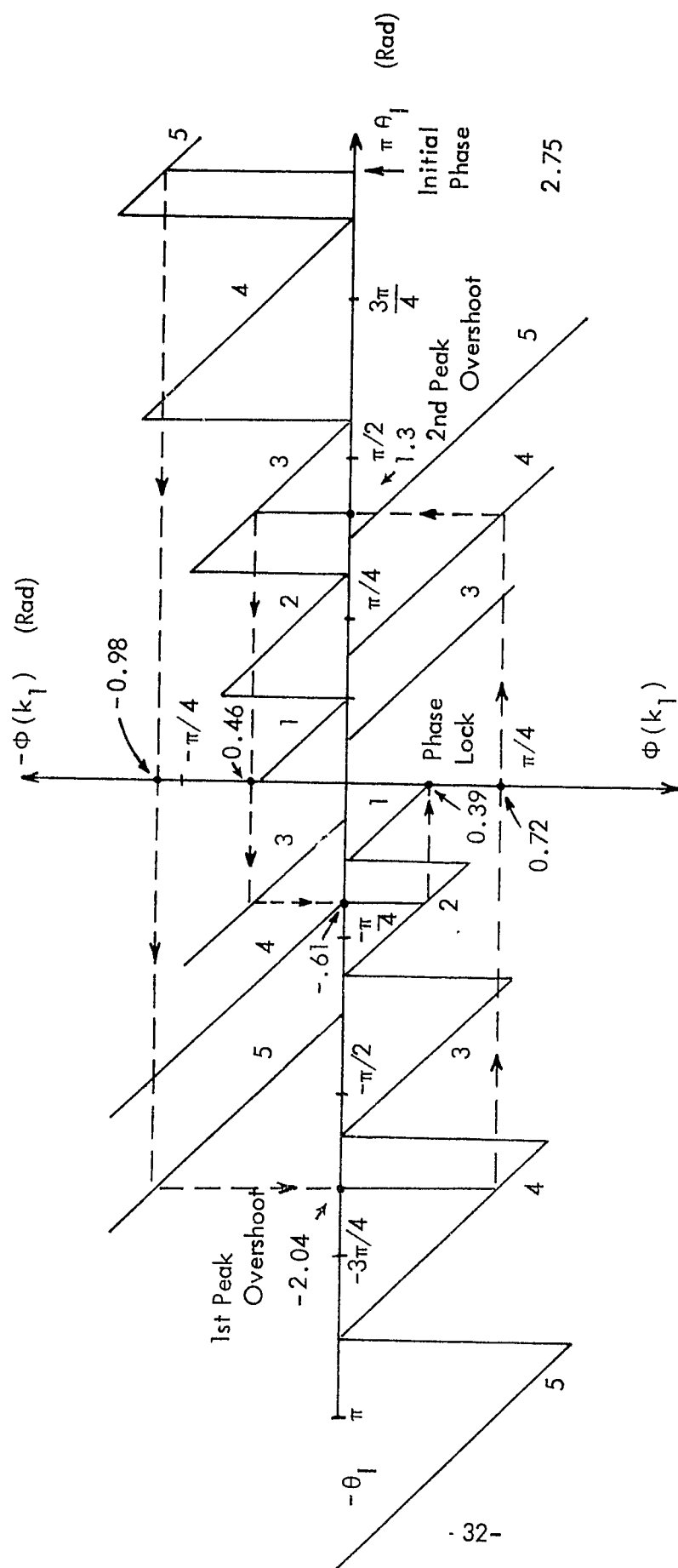


Figure 3-8. Determination of Specified Response Parameters for an Initial Phase Offset.

to find that the second zero crossing occurs four samples after the peak overshoot and has value 0.72 radians. The value of the second peak overshoot is seen to be 1.3 radians. The value of the second peak overshoot is then used to find the third zero crossing and the third peak overshoot value. This process is just continued in a clockwise manner until only two samples are required for the occurrence of the i -th zero crossing at which time phase lock has occurred. Thus the phase function characteristics for any initial phase offset can easily be determined by Figure 3-8.

Consider next the case where the input signal differs by a constant frequency from the quiescent frequency of the second order DPLL with $\theta_1(k) = 0$. Since for this case the condition of (3-31) is not assured during loop phase acquisition the phase error difference equation is given by

$$\begin{aligned} \phi(k+1) - \phi(k) = & 2\pi (1 - \omega/\omega_0) - \Delta_1 \frac{\omega}{\omega_0} \operatorname{sgn} [\sin \Phi(k)] \quad (3-60) \\ & - \omega/\omega_0 \Delta_2 \sum_{i=1}^k \operatorname{sgn} [\sin \Phi(k)] \end{aligned}$$

with $\phi(0) = \theta_1$. The inclusion of the constant term for the frequency difference and the sine function complicates (3-60) to the point that even a partial solution as was performed for the case of zero frequency difference is not possible. However, it is possible to study the stability of the second order DPLL by means of an incremental phase plane portrait. In the incremental phase plane, $\phi(k+1) - \phi(k)$ is plotted as the ordinate and $\phi(k+1)$ is plotted as the abscissa while k is a variable parameter. Then for any k , $[\phi(k+1), \phi(k+1) - \phi(k)]$ describes the state of the system and a stable region of operation is found when, for any given initial conditions,

$$[\phi(k+1), \phi(k+1) - \phi(k)] \rightarrow [2\pi i, 0.0] \quad i = 0, \pm 1, \pm 2 \dots (3-61)$$

as $k \rightarrow \infty$

For the case of $\Delta_1 = \Delta_2 = \pi/32$ and $\theta_1 = \pi/4$ incremental phase plane portraits have been plotted in Figures 3-9 and 3-10 for various values of ω/ω_0 . From Figure 3-9 it is seen that for $\omega/\omega_0 < 1$, but sufficiently large, that it is possible to achieve phase lock without cycle slipping. However, as ω/ω_0 decreases, the loop will achieve phase lock after slipping an increasing number of cycles. Notice that as the number of cycles slipped increases the length of time required to achieve phase lock also increases. Similarly, from Figure 3-10 it is seen that for $\omega/\omega_0 > 1$, but sufficiently small, that it is possible to achieve phase lock without cycle slipping but as ω/ω_0 increases the number of cycles slipped prior to phase lock increases.

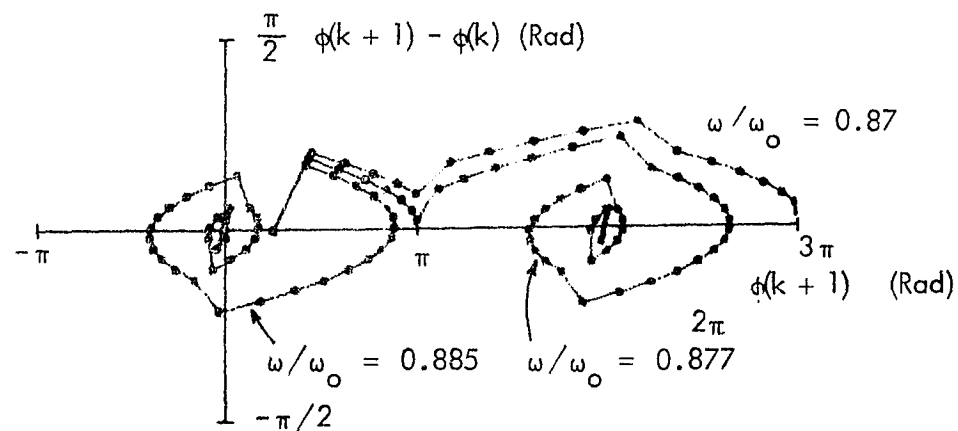


Figure 3-9. Incremental Phase Plane Portrait for $\theta_1 = \pi/4$ and $\omega/\omega_0 < 1$.

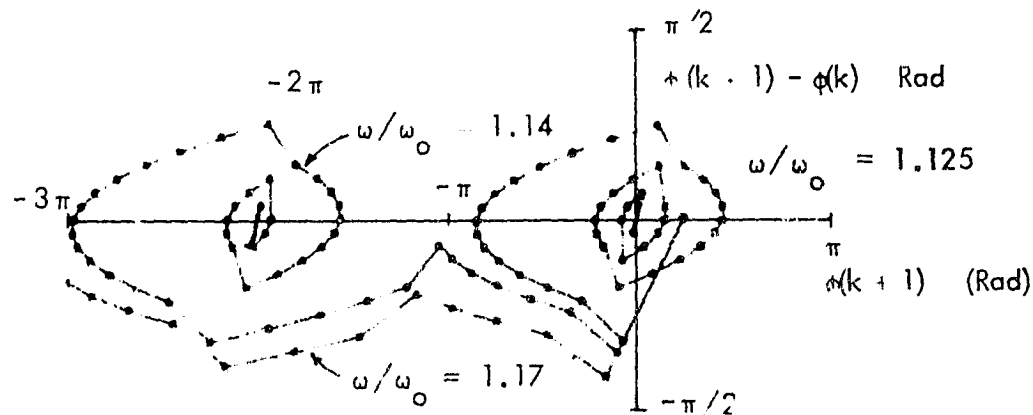


Figure 3-10. Incremental Phase Plane Portrait for $\theta_1 = \pi/4$ and $\omega/\omega_0 > 1$.

For some applications it is necessary that phase lock be achieved in minimum time so that it is necessary that no cycle slipping occur during phase acquisition. Consider first the case of $\theta_1 > 0$ and $\omega/\omega_0 < 1$, so that the phase error can be expressed as

$$\begin{aligned} \phi(k+1) = & \theta_1 + 2\pi(1 - \omega/\omega_0)k - \omega/\omega_0 \Delta_1 k \\ & - \omega/\omega_0 \Delta_2 \frac{k(k+1)}{2} \end{aligned} \quad (3-62)$$

up to either the first zero crossing of the phase error function or the first cycle slip since $\text{sgn}[\sin \phi(k)] = 1$ until either of these occurrences. Equation (3-62) can be rewritten as

$$\phi(k+1) = \theta_1 - \Delta_1' k - \Delta_2' \frac{k(k+1)}{2} \quad (3-63)$$

where

$$\Delta_1' = -2\pi + \omega/\omega_0 (2\pi + \Delta_1) \quad (3-64)$$

$$\Delta_2' = \omega/\omega_0 \Delta_2 \quad (3-65)$$

and for a cycle slip to occur

$$\phi(k+1) \geq \pi \quad (3-66)$$

From (3-63) and (3-66)

$$k^2 + \frac{2\Delta_1 + \Delta_2}{\Delta_2} k - \frac{2}{\Delta_2} (\theta_1 - \pi) \leq 0 \quad (3-67)$$

Since k is the sample number of the occurrence of a cycle slip during phase acquisition, k must be a positive, real value. Since k must be real then

$$\frac{2\Delta_1 + \Delta_2}{2\Delta_2} + \frac{2}{\Delta_2} (\theta_1 - \pi) \geq 0 \quad (3-68)$$

and, since $\theta_1 < \pi$, for k to be positive then,

$$\frac{2\Delta_1 + \Delta_2}{2\Delta_2} \leq 0 \quad (3-69)$$

For (3-68), the worst case for cycle slip occurs as $\theta_1 \rightarrow \pi$, therefore if $\theta_1 = \pi - \delta$

and $\Delta_1 = \Delta_2 = \pi/32$ the constraint of (3-68) gives

$$\omega/\omega_0 \geq 0.971 \text{ for } \delta = 0.01 \quad (3-70)$$

while the constraint of (3-69) gives

$$\omega/\omega_0 \geq 0.977 \quad (3-71)$$

Therefore, k will be a positive real value for indicating a cycle slip will occur for

all values of $\omega/\omega_0 \geq 0.971$. A similar procedure can be followed for

$-\pi + \delta \leq \theta_1 \leq 0$ and $\omega/\omega_0 \geq 1$ which for the same case of $\Delta_1 = \Delta_2 = \pi/32$

will give a cycle slip for

$$\omega/\omega_0 \geq 1.03, \delta \approx 0.01 \quad (3-72)$$

Thus once Δ_1 and Δ_2 are defined, it is possible for the 2nd order DPLL to define the frequency offset range over which phase acquisition can occur without cycle slipping.

CHAPTER IV

FIRST ORDER DPLL WITH ADDITIVE NOISE INPUT

A. Introduction. In the preceeding chapter partial response characteristics for a first- and second-order DPLL with ideal input were developed. As for the case of an ideal input, little has appeared in the literature concerning the characteristics of DPLL operation with a noisy input. For the most part any analysis that has appeared in the literature for stochastic inputs has involved linearizing assumptions on the DPLL's operation. Two notable exceptions have been the DPLL configurations studied by Cessna and Levy [5] and Holmes [6] where random walk techniques were used to determine the statistical characteristics of the loop. For the DPLL of reference [5], the first order loop utilized resettable low pass digital filters in the phase correction path while the DPLL in reference [6] did not utilize a filter. A DPLL configuration that uses a non-resetting filter is analyzed in the following without linearizing assumptions using a Markov chain model that achieves the same statistical performance of that described in [5] but with less physical complexity.

B. First-Order DPLL Configuration. The specific loop configuration considered here is shown in its implementation form in Figure 4-1 and is a slightly modified version of the generalized DPLL model shown in conceptual form in Figure 3-1 with $\Delta_2 = 0$. This loop configuration is that of the Ohio University MAPLL [7] with the exception that the loop is assumed to operate continuously instead of in a gated manner. Assuming for the moment that $M=1$ in Figure 4-1, then this loop implementation is easily seen to be identical to the model of Figure 3-1 with $\Delta_2 = 0$ since at each sample of the input signal the phase of the reference clock will either be advanced or retarded by $\Delta_1 = \frac{2\pi}{N}$ radians depending on whether the reference leads or lags the input signal.

An improvement can be made in the loop's operation for a fading input by the addition of the divide-by M up/down counter preceeding the divide-by N up/down counter. It is obvious that to change the phase of the reference clock it is necessary for the divide-by M counter to cycle through its M distinct states to either an overflow or underflow condition. Thus, while the phase output takes on N distinct states, the loop itself has M x N distinct states. Further, the divide-by M counter constitutes a digital low-pass filter whose transfer function is found in Appendix D as,

$$H(j\omega) = \frac{1}{M} \frac{\sin M \frac{\omega}{\omega_c} \pi}{\sin \omega \pi / \omega_c} e^{-j \frac{\omega \pi}{\omega_c} (M-1)} \quad (4-1)$$

It is seen from the plots of the magnitude-squared function of (4-1) in Appendix D that as M increases, the bandwidth of the filter decreases.

A state diagram for the loop is given in Figure 4-2 where the values of p_i and q_i are the probabilities associated with the indicated state changes. Several things are worth noting about this state diagram. First, for any present state, when a new sample is taken a new state will result; and second, the new state will always be adjacent to the previous state. Also, for a given reference clock state the transitions occur uniformly with time but, following a transition from one reference clock state to another, the time interval to the next sample is either longer or shorter than the time interval between the previous two samples depending on whether the reference clock phase was advanced or retarded as it passed from the previous reference clock state to the present reference clock state.

If the phase samples are independent and the non-uniform sampling interval is ignored, then the DPLL given by the state diagram of Figure 4-2 can be approximated by a first-order Markov chain, the properties of which are outlined in Appendix A.

If a_{ij} is defined as the probability that the loop is in state s_{ij} , then the system of equations describing the loop probabilities is given as,

[illegible]

Or,

$$[P][a] = [a] \quad (4-3)$$

where $[P]$ is called the matrix of transition probabilities. The non-trivial solution to this system of homogeneous, linear equation is the one that satisfies the total probability relationship,

$$\sum_{i=1}^N \sum_{j=1}^M a_{ij} = 1 \quad (4-4)$$

and, from the characteristics of Markov chains, represents the steady state probability for the loop states. That is, at any given instant of time the probability of observing the loop in state s_{ij} is given by a_{ij} .

The steady state probabilities for the reference clock states, s_i are given by,

$$s_i = \sum_{j=1}^M a_{ij} \quad i=1, 2, \dots, N \quad (4-5)$$

Once the steady state reference clock state probabilities are known, it is possible to find the variance of the reference clock phase from

$$\sigma_{\Phi}^2 = \frac{1}{N} \sum_{i=0}^{N-1} s_{i+1} \left[\frac{\pi}{N} (1 - N + 2i) \right]^2 \quad (4-6)$$

where it has been assumed that the phase of the input signal is a constant 0. radians corrupted by noise.

Another quantity useful for the evaluation of a DPLL with a fading input is the mean time to lock-up for some initial phase offset. This can be evaluated by considering the loop transient response in terms of the classical ruin problem (Feller [8]) for a random walk. The mean time to lock for an initial phase offset then would

be equivalent to a determination of the expected duration of the game in the classical ruin problem. If $T_{oi,j}$ is defined as the expected time required to reach a minimum phase error when the initial state of the loop is s_{ij} then the $T_{oi,j}$ must satisfy the difference equation,

$$T_{oi,j} = 1 + p_i T_{oi,j-1} + q_i T_{oi,j+1} \quad (4-7)$$

for $J \neq 1, M$ and with the boundary conditions

$$T_{ok,j} = T_{k+1,j} = 0 \quad j = 1, 2, \dots, M \quad (4-8)$$

where reference clock states k and $k+1$ represent the minimum phase error states.

For the input signal assumed to be 0. radians then the system of non-homogeneous, linear simultaneous equations results,

C. Matrix of Transition Probabilities.

Consider the case where the input to the DPLL is of the form;

$$s_r(t) = s(t) + n(t) \quad (4-10a)$$

$$= A_c \cos(\omega_c t + \psi) + x(t) \cos \omega_c t + y(t) \sin \omega_c t \quad (4-10b)$$

where A_c is the carrier amplitude and $x(t)$ and $y(t)$ are zero mean independent gaussian distributed random processes of bandwidth B and variance $\sigma_x^2 = \sigma_y^2 = \sigma^2$. That is, the input to the DPLL is some signal plus narrowband noise. The input can also be written in the form:

$$s_r(t) = x'(t) \cos \omega_c t + y'(t) \sin \omega_c t \quad (4-11)$$

where

$$x'(t) = x(t) + A_c \cos \psi \quad (4-12a)$$

$$y'(t) = y(t) + A_c \sin \psi \quad (4-12b)$$

The positive going zero crossing is always assumed to be the correct phase of the signal $s(t)$. That is, the reference clock is always assumed to be in phase lock with the signal $s(t)$, and the error signal generated by the phase detector is used to tell the loop differently. Therefore, the loop always assumes the samples of the incoming signal occur at:

$$\omega_c t_i = (1 + 2i) \pi / 2 \quad i = 0, 1, 2, \dots \quad (4-13)$$

so that

$$\cos \omega_c t = 0 \quad (4-14a)$$

$$\sin \omega_c t = 1 \quad (4-14b)$$

giving

$$s_r(t) = y(t) + A_c \sin \psi \quad (4-15)$$

where ψ is the phase difference between $s(t)$ and the reference clock. Note that as the loop approaches lock, ψ approaches 0. Also, the reference clock can take on only N distinct values so that ψ too can take on only N distinct values ψ_i , $i = 1, 2, \dots, N$. From earlier $y(t)$ is a gaussian distributed random process so that the probability density function for $s_r(t)$ is:

$$p(s_r) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2} \frac{(y + A_c \sin \psi_i)^2}{\sigma^2}} \quad (4-16)$$

From the state diagram of Figure 3, the p_i 's are the probability that the sampled value of $s_r(t)$ is less than zero and can be found from:

$$p_i = \int_{-\infty}^0 p(s_r | \psi_i) ds_r \quad (4-17a)$$

$$= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^0 e^{-\frac{1}{2} \left[\frac{y + A_c \sin \psi_i}{\sigma} \right]^2} dy \quad (4-17b)$$

$$= \text{probability that } s_r(t) \leq 0 \quad (4-17c)$$

Also,

$$q_i = 1 - p_i \quad (4-18a)$$

$$= \text{probability that } s_r(t) > 0 \quad (4-18b)$$

D. Application of the Markov Chain Model. The phase error and transient response of the DPLL given in Figure 4-1 can be determined from the solution of (4-2) and (4-9), respectively, once the values of M and N have been specified. To perform the solution to these two sets of equations, the three Fortran computer programs given in Appendix B were written. The first program, PBSTGEN, is used to set up the matrix of transition probabilities for the systems of (4-2) and (4-9) once

the values of M and N are given. The second program given in Appendix B, PBDPL2, is used to solve for the steady state probabilities of the loop states and from this calculates the steady state probabilities of the reference clock states and finally the variance of the phase error for specified input signal-to-noise ratios. The final program given in Appendix B, PBDPL2T, is used to determine the solution to the transient response system of equations given in (4-9). Both PBDPL2 and PBDPL2T use a successive approximation technique to determine the solution to the system of equations. Note, all three programs were written to be used both for the first-order DPLL considered in this chapter and also for the second-order DPLL which will be given in the next chapter.

The steady state probabilities for the reference clock states were found for various values of M and signal-to-noise ratio with $N=64$ and are plotted in the terms of the phase error probability density function in Figures 4-3 and 4-4. The value of $N = 64$ was chosen to be consistent with the application to an Omega navigation receiver as outlined in [7]. Note that for all cases, the value of N and M will be assumed to be a power of two so that a practical realization of the counters can be achieved with a standard binary counter. In Figure 4-3, the envelope of the probability density function for the phase error is shown for $M=1$ and signal-to-noise ratios of 20., 0.0, -20., and 40. dB. Notice that as the signal-to-noise ratio decreases, the envelope of the density function approaches that of a uniform phase error distribution as would be expected. In Figure 4-4, the envelope of the density function is shown for a constant signal-to-noise ratio of -20. dB while M takes on values 1, 2, 4, and 8. From this plot the effects of the digital low-pass filter (the divide-by M counter) can be seen since as the bandwidth of the filter decreases, the den-

sity functions tends to flatten.

Once the steady state probabilities of the reference clock states are determined the phase error variance can be easily calculated. In Figure 4-5, the standard deviation of the phase error is plotted for $M=1, 2, 4$, and 8 versus noise-to-signal ratio. For low values of noise-to-signal ratio these curves are asymptotic to a value fixed by N , the quantization level of the reference clock. For high noise-to-signal ratios, these curves are asymptotic to the standard deviation of a uniform phase distribution. Note also that as the value of M increases by a power of two that there is an approximate 3.0 dB increase in the performance of the loop.

The mean time to lock in terms of the number of samples required to achieve phase lock for some initial phase offset was determined for $M=1, 2$, and 4 for various values of signal-to-noise ratio. Note that phase lock following some initial phase offset is defined as the first occurrence of the minimum phase error condition as determined from the solution of (4-9). Since the phase of the reference clock is assumed to take on values from $-\pi$ to π , a minimum phase error condition would be reached for $M=2$ when the loop reached states $s_{32,1}$, $s_{32,2}$, $s_{33,1}$, or $s_{33,2}$ for $N=64$ and the input signal a constant 0.0 radians. However, for the case considered here, a more stringent lock condition was imposed in that only states $s_{32,2}$ and $s_{33,1}$ were considered to constitute phase lock. For this case the boundary conditions for (4-9) become $T_{o32,2}$ and $T_{o33,1}$ equal to zero. Similarly, for the case of $M=4$, only the states $s_{32,4}$ and $s_{33,1}$ were considered to constitute phase lock. The solution to (4-8) under the above conditions are plotted in Figures 4-6, 4-7, and 4-8. As expected, for a given initial phase offset, the mean number of samples to the first occurrence of a phase lock condition increases as the signal-

to-noise ratio decreases for all three plots. Notice that for a 20. dB signal-to-noise ratio that the expected time to lock is very nearly equal to the time required for lock under ideal conditions. A cross comparison between Figures 4-6, 4-7, and 4-8 indicates that the expected time to achieve phase lock increases as the value of M increases as would be required since more loop states must be traversed to achieve phase lock.

In addition to the theoretical values, Figures 3-5 through 4-8 also include experimental data taken from a hardware realization of the first-order DPLL. This hardware realization will be discussed in Chapter VI.

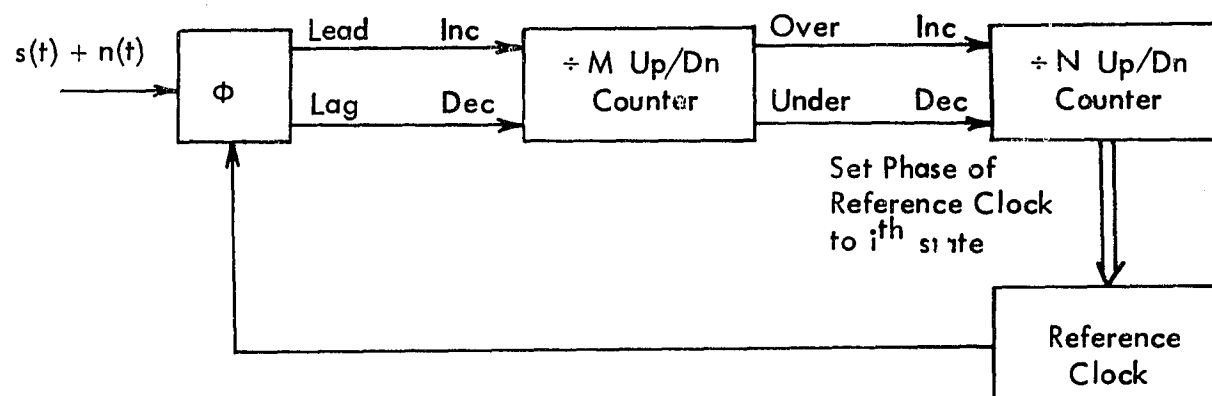


Figure 4-1. First-Order DPLL Implementation.
 Note: i is Present Value of $\div N$ Up/Dn Counter.

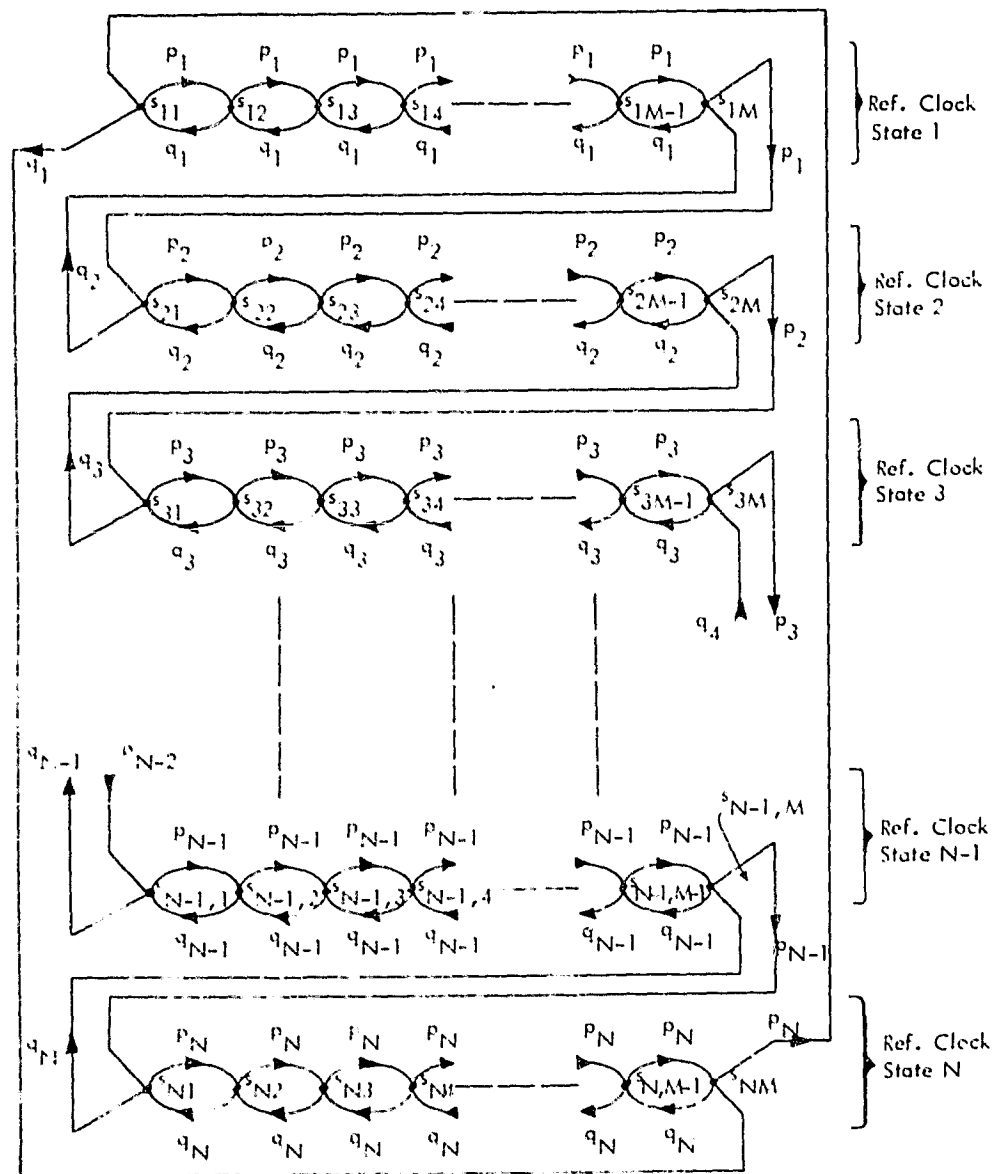


Figure 4-2. First Order DPLL State Diagram .

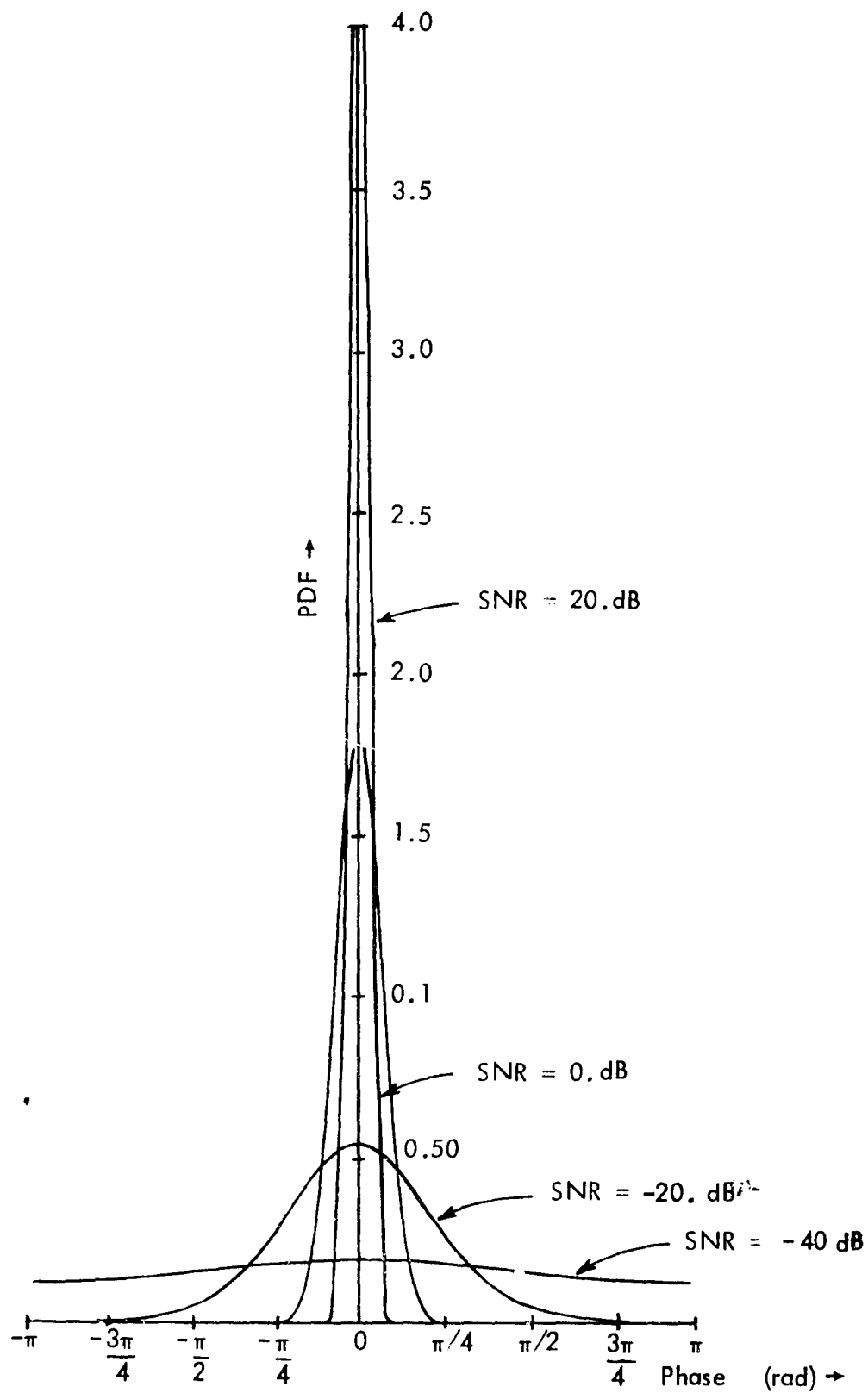


Figure 4-3. Probability Density Function of Phase Error, $N=64$, $M=1$.

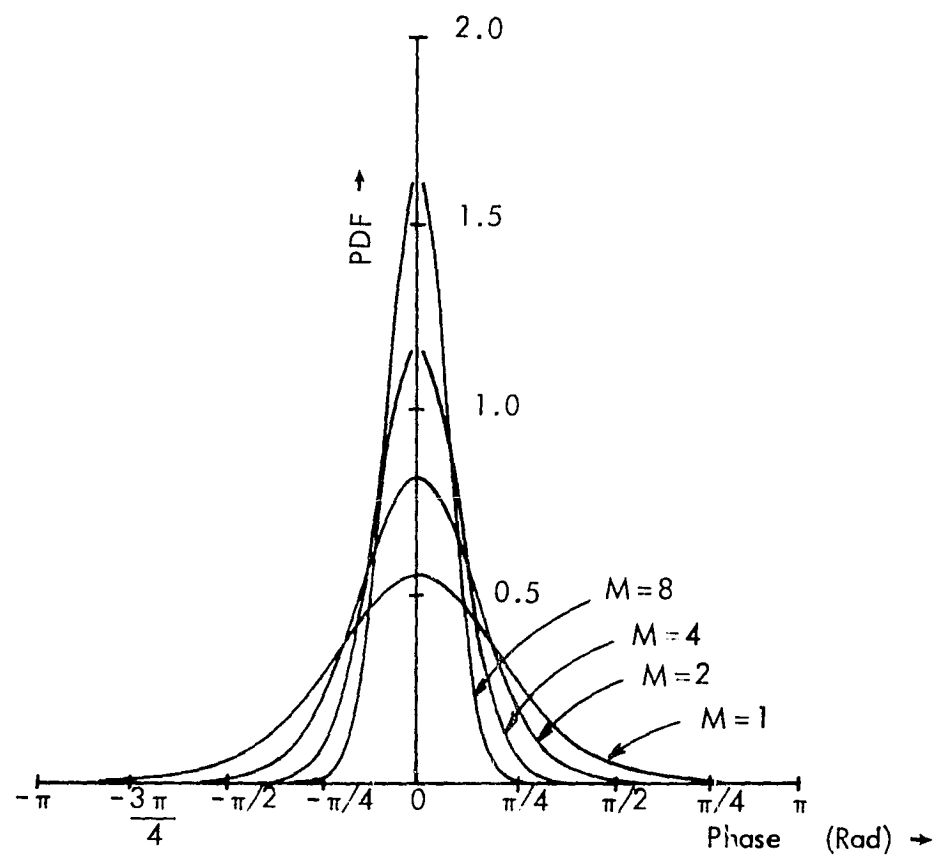


Figure 4-4. Probability Density Function of Phase Error, $N = 64$, $\text{SNR} = 20 \text{ dB}$.

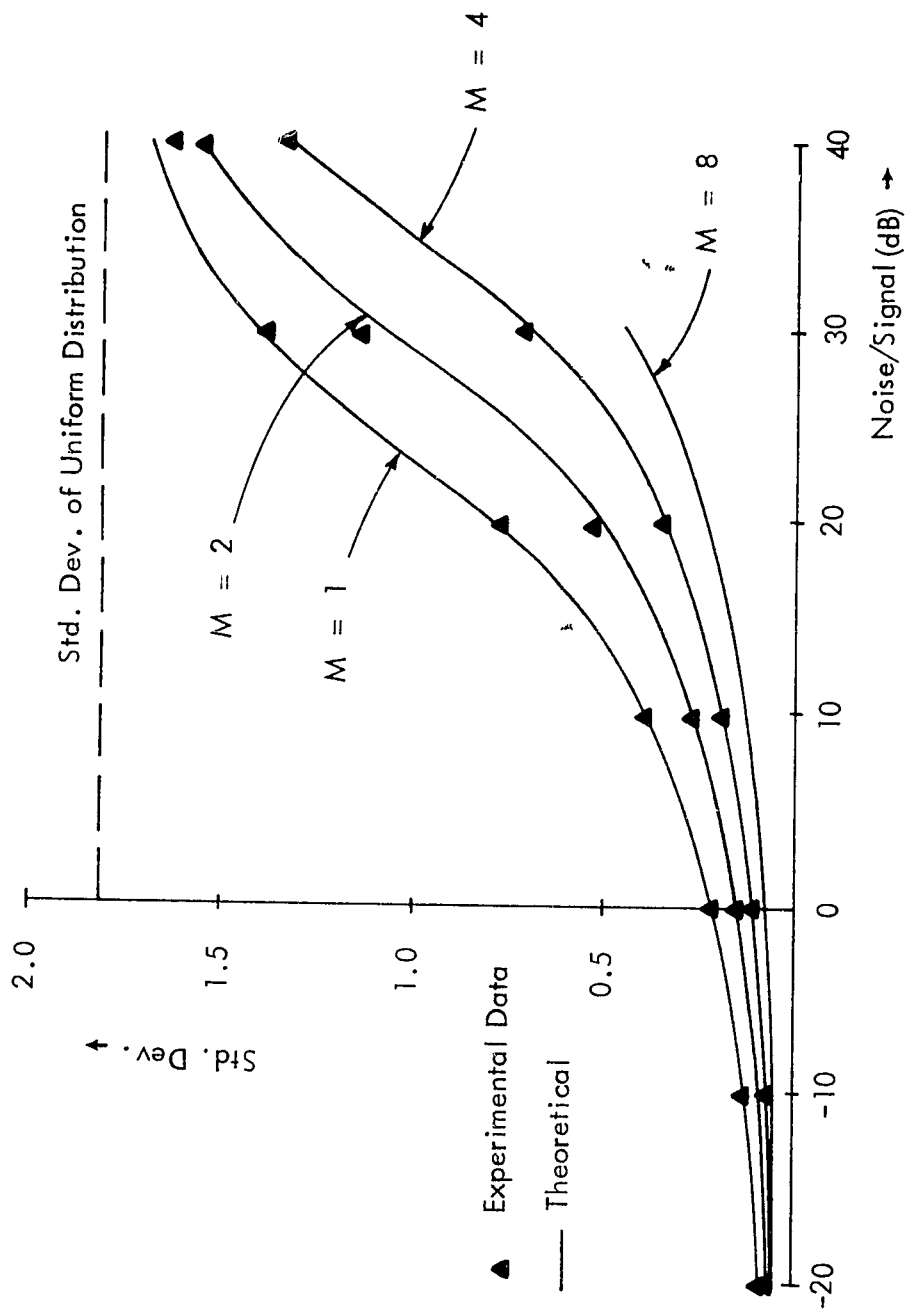


Figure 4-5. Standard Deviation of Loop Phase Error Vs. Noise to Signal Ratio, $N = 64$.

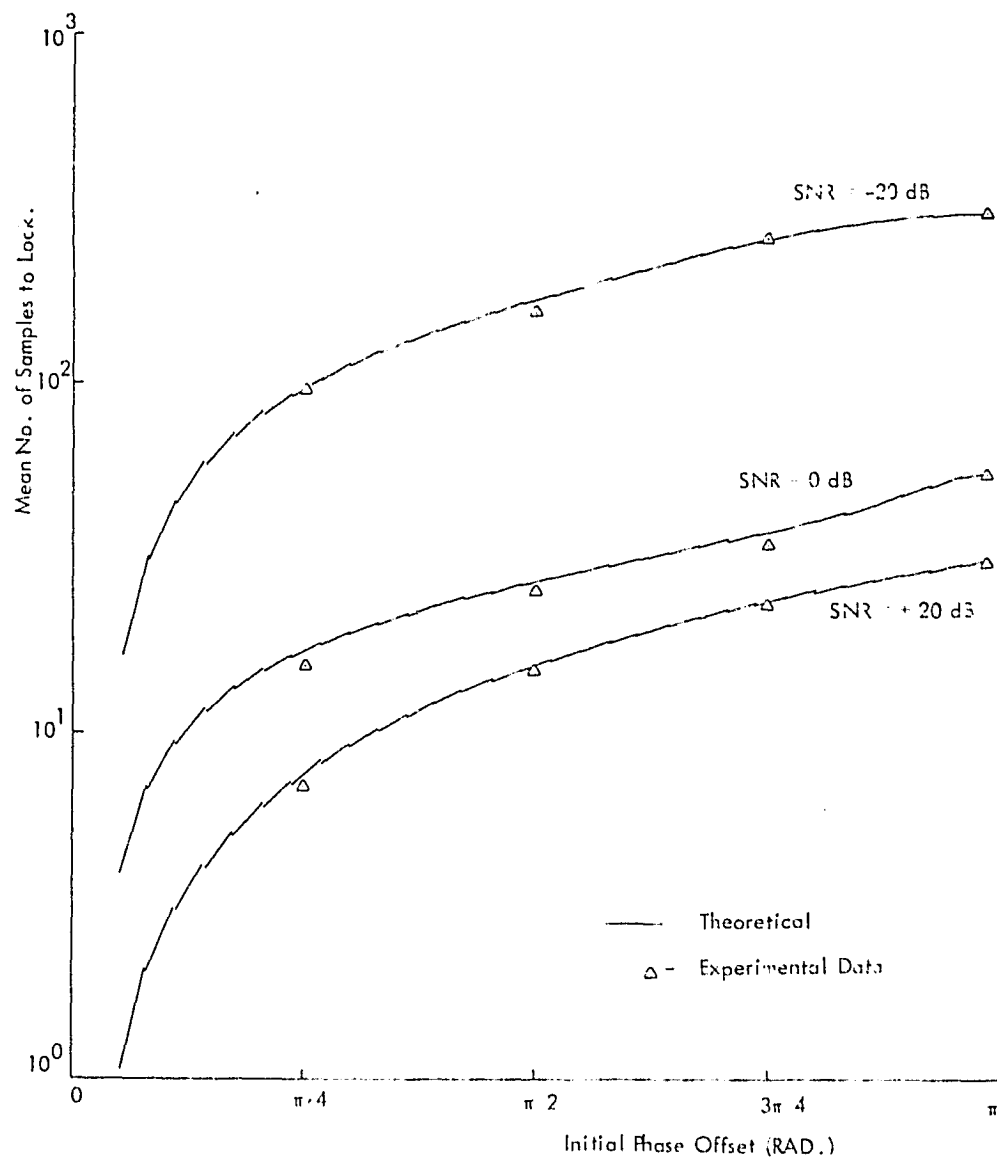


Figure 4-6. Loop Transient Response
 $M=1$, $N=64$.

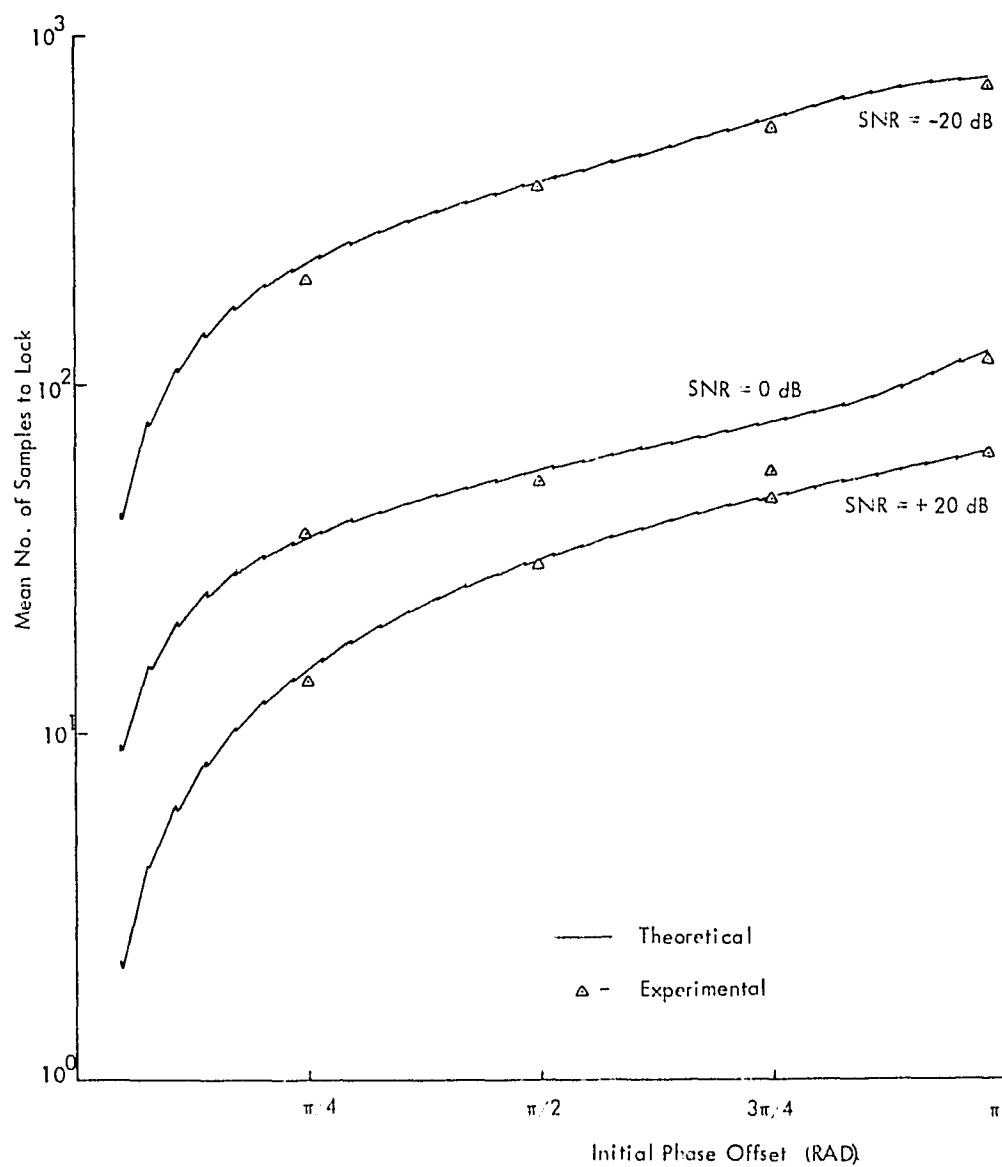


Figure 4-7. Loop Transient Response
 $M=2, N=64$.

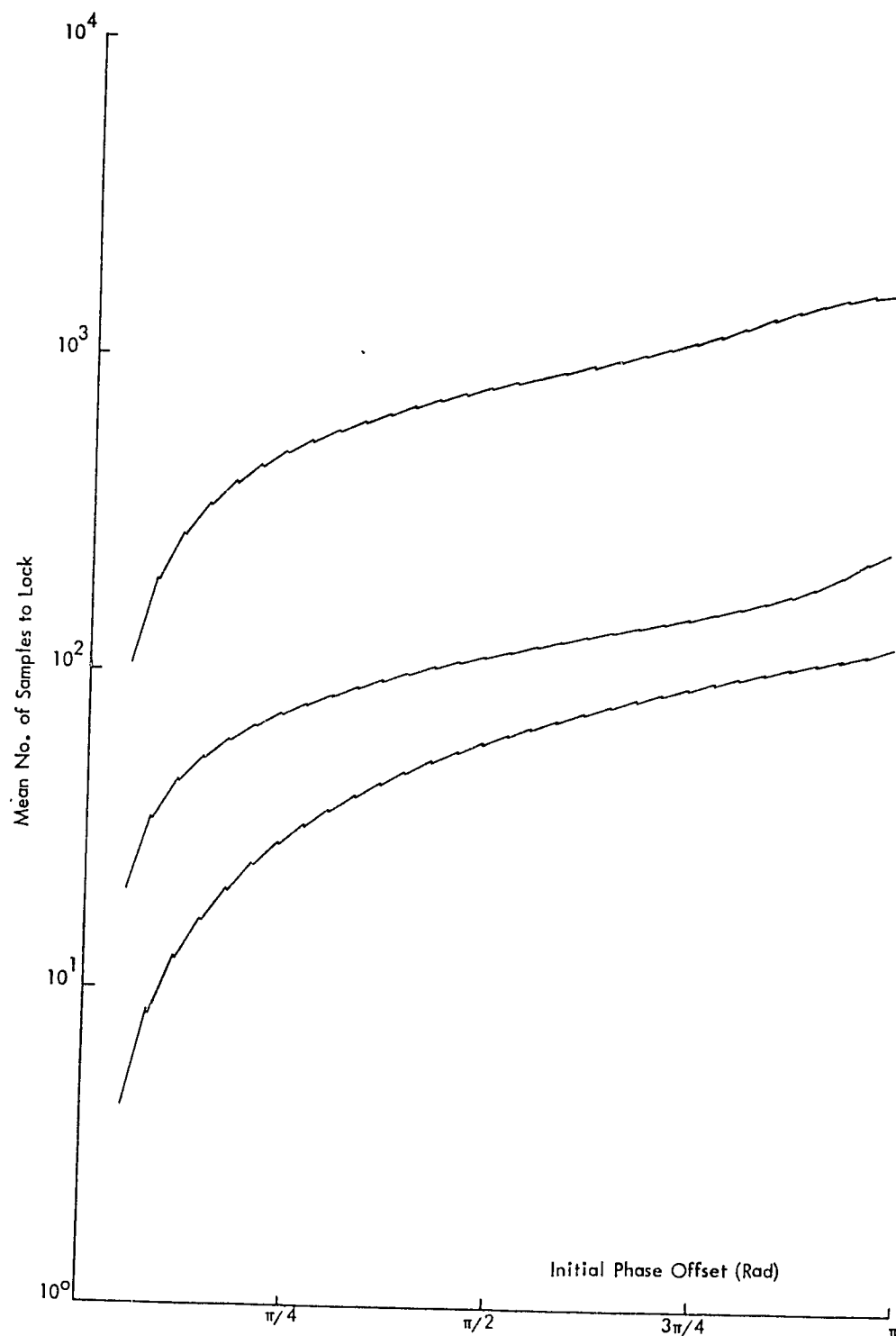


Figure 4-8. Mean No. of Samples to Lock, $N = 64$, $M = 4$.

V. SECOND ORDER DPLL WITH ADDITIVE NOISE INPUT

A. Introduction. Second order DPLL configurations have been analyzed by Holmes and Tegenelia [13] and Weinburg and Liu [4] for an additive white gaussian noise input. However, in the case of reference [13], the loop was modeled as a linear system and then analyzed by classical techniques. The results presented for this linear model showed close agreement with experimental data for signal-to-noise ratios greater than 0.0 dB. In reference [4], the steady state phase error for the second order DPLL was determined from a solution of the Chapman-Kolmogorov equation in the z-domain under the assumption of small values of steady state phase error. Again, the usefulness of this solution is limited by the input signal-to-noise ratio. In this chapter, the author presents a method of analysis for a second order DPLL that is not limited by the input signal-to-noise ratio.

In the previous chapter, the analysis of a first order DPLL was performed by modeling the loop as a first order Markov chain. In the following, the idea and methods used in the previous chapter will be modified to allow a similar approach to be used for the analysis of second order DPLL's. It will be shown that a second order DPLL can be modeled as a first order Markov chain with alternatives and that these alternatives themselves can be thought of as states in a first order Markov chain. The steady state distribution of the Markov chain alternatives can be determined and from this distribution it is possible to find the steady state phase error of the DPLL. The transient response of the loop is also determined in a similar manner.

B. Second Order DPLL Configuration. The second order DPLL configuration considered in this paper is shown in its implementation form in Figure 5-1. The loop is identical to that of Figure 4-1 with the exception of the addition of the divide-by

L up/down counter and the K-bit binary adder where $K = \log_2 (M \times N)$. The inclusion of the divide-by L counter provides the $\Delta_2 \sum_{i=1}^k X(i)$ function of the generalized model of Figure 3-1. Note however that for a practical implementation, the divide-by L counter cannot be allowed to either overflow or underflow since this would have the effect of resetting the summation value to zero. Therefore, the divide-by L counter is structured so that it will saturate at values of $\pm L$.

Operation of the loop is as follows. The input signal is sampled at the positive-going zero crossing of the reference clock to determine whether the reference clock leads or lags the input signal. If a phase lead is detected, the divide-by M and divide-by L counters are incremented by one and if phase lag is detected, the divide-by M and divide-by L counters are decremented by one. Following this, the value of the divide-by M and divide-by N counter combination is added to the value of the divide-by L counter and the result loaded into the divide-by M and divide-by N counter combination. Finally, the phase of the reference clock is updated to reflect the value contained in the divide-by N counter after the load has occurred.

Referring to Figure 3-1, the value of Δ_1 will in this case still be given by $\Delta_1 = \frac{2\pi}{N}$ and the value of Δ_2 is determined by the bit in the divide-by M and divide-by N counter combination to which the least significant bit of the divide-by L counter is added. For example, if the binary adder is configured so that the least significant bit of the divide-by L counter adds to the least significant bit of the divide-by N counter then $\Delta_1 = \Delta_2$. However, if the least significant bit of the divide-by L counter adds to the most significant bit of the divide-by M counter, then $\Delta_1 = 2^M \Delta_2$. Note that this leads to a more limited relationship for Δ_1 and Δ_2

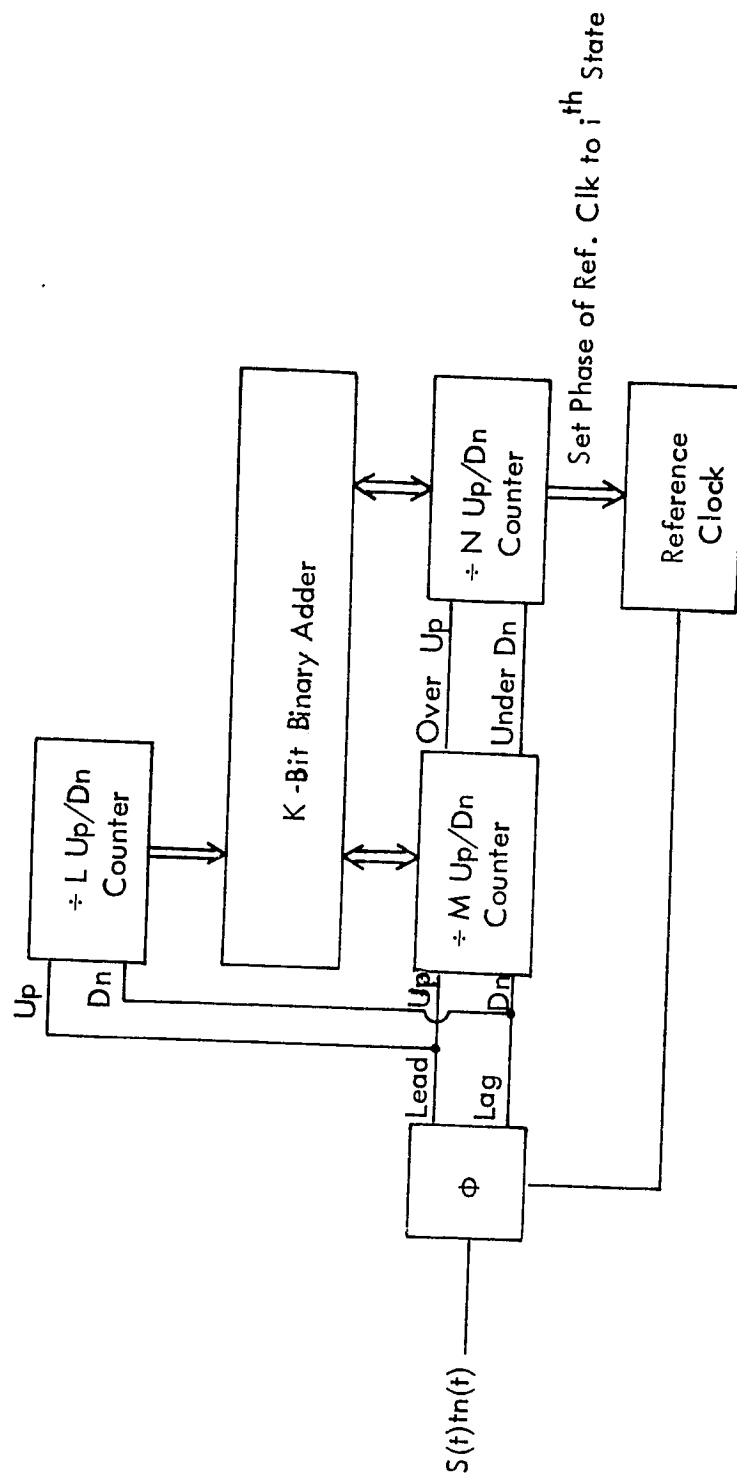


Figure 5-1. Second Order DPLL Implementation.

Note: i is Present Value of $\div N$ Up/Dn Counter.

than given in (3-42) since for this configuration,

$$\Delta_1 = 2^i \Delta_2 \quad i = 1, 2, \dots \quad (5-1)$$

Note also that a first order DPLL results from this configuration if the divide-by L counter is configured to saturate in its zero state.

For the first order DPLL, the loop states were defined by the combination of the divide-by M and divide-by N counters giving a total of $M \times N$ loop states. For the second order DPLL, the loop states are defined by the value contained in the combination of the divide-by M and divide-by N counters following the loading of the K-bit binary adder output so that once again there are a total of $M \times N$ loop states. However, for the second order DPLL the loop state transitions are dependent not only on the present output of the phase detector but also upon the value contained in the divide-by L counter. Thus the value of the divide-by L counter can be thought of as providing alternative loop state to loop state transitions for a given phase detector output. The loop can then be modeled as a first order Markov chain with alternative state transition vectors. The characteristics of the Markov chain with alternatives is discussed in Appendix A.

C. Markov Chain Model of the Second Order DPLL. Use of the Markov chain with alternatives for the modeling of the second order DPLL configuration considered here will be shown by means of an example. For this, consider the DPLL of Figure 5-1 with $M = 2$, $N = 4$, and $L = 3$ so that there are eight loop states which define the states of a first order Markov chain. However the divide-by L counter can take on seven values so that for each loop state there are seven possible alternative loop state transitions. A state diagram of the Markov chain model for the DPLL under consideration is shown in Figure 5-2. In this figure, the loop states are situ-

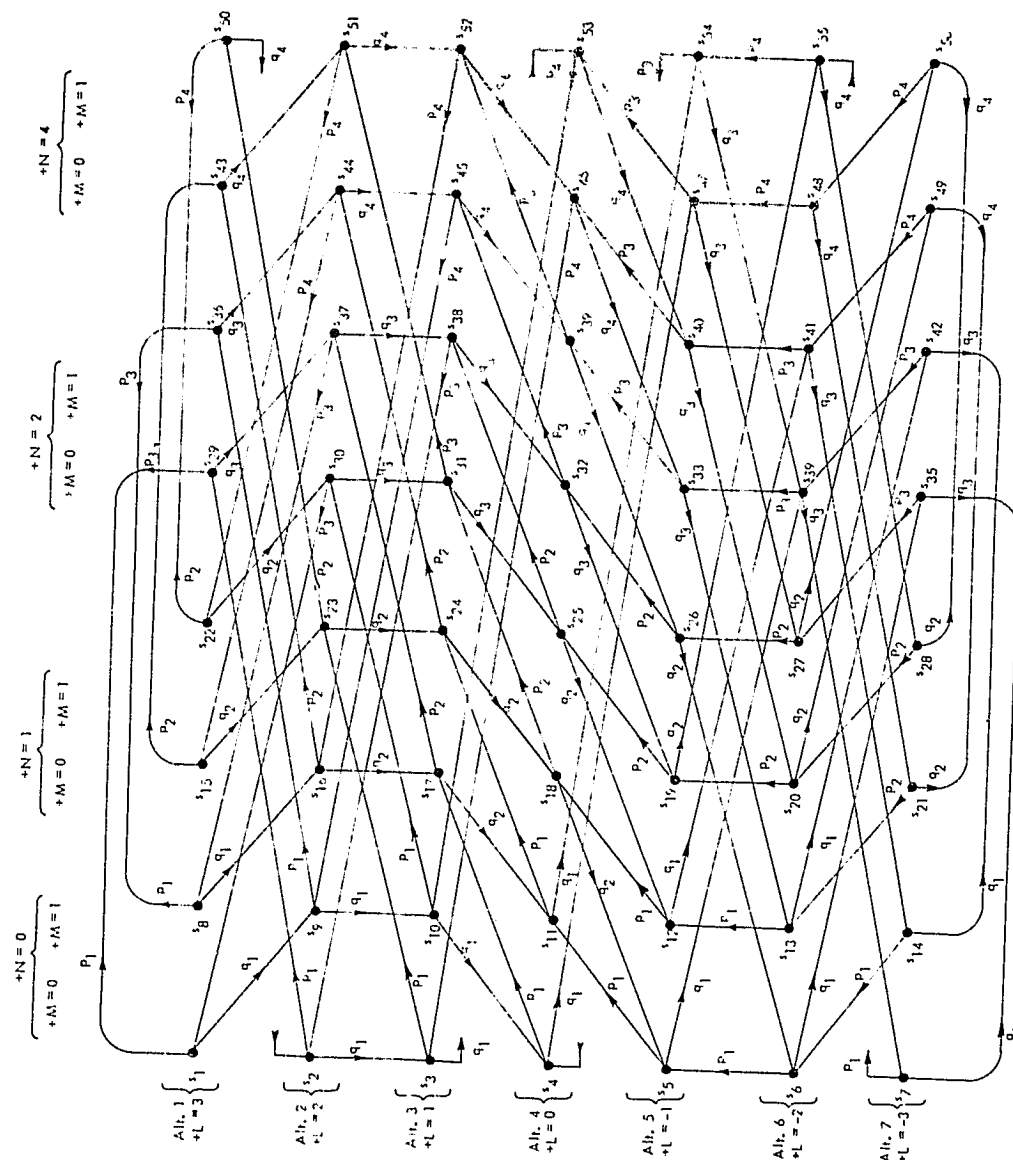


Figure 5-2. DPLL State Diagram, $N = 4$, $M = 2$, $L = 3$.

ated horizontally while the alternatives associated with each loop states are situated perpendicularly. Thus the top row labeled alternative 1 gives the state transitions for the divide-by L counter having a value +3. Similarly alternative 2 applies for the divide-by L counter having value 2, alternative 3 is for the divide-by L counter having value 1 and so on. The loop states are plotted such that the first column is for a reference clock state of one with the value of the divide-by M counter equal to zero while the second column is for reference clock state one with the value of the divide-by M counter equal to one. Similarly, the third column is for reference clock state two with a value of zero contained in the divide-by M counter and so on.

Since there are eight possible loop states with each loop state having seven possible alternative actions, there are a total of 56 alternative state vectors in the matrix of transition probabilities as given in Appendix A. However, for each loop state, the alternative vector to be used is uniquely defined by the value of the divide-by L counter so that the alternative vectors themselves can be thought of as states in a first order Markov chain. That is, instead of considering the transition from loop state to loop state, the transitions from loop alternative to loop alternative are considered. Thus in the state diagram of Figure 5-2, the loop alternatives are successively numbered s_1, s_2, \dots, s_{56} and the possible transition from loop alternative to loop alternative is assigned a probability as indicated by the directed arrows. The values of the indicated probabilities are dependent only upon the reference clock state associated with each loop alternative and the method of determining their value was discussed in Chapter IV.C.

To show that the state diagram of Figure 5-2 will achieve a phase lock condition, consider the case of an ideal input signal. For this case the transition probabilities as given by (4-16b) and (4-17a) are,

$$p_1 = p_2 = q_3 = q_4 = 1 \quad (5-2)$$

and

$$p_3 = p_4 = q_1 = q_2 = 0 \quad (5-3)$$

If the loop is assumed to initially be in alternative state s_4 , then on successive samples of the input the path traced through the state diagram will be

$$s_4 \rightarrow s_{17} \rightarrow s_{37} \rightarrow s_{38} \rightarrow s_{32} \rightarrow s_{19} \rightarrow s_{25} \rightarrow s_{38}$$

so that when phase lock is achieved, the loop will continuously cycle through the closed path

$$s_{38} \rightarrow s_{32} \rightarrow s_{19} \rightarrow s_{25} \rightarrow s_{38} \dots$$

Inspection of Figure 5-2 will show that the same state cycle will be obtained for any initial alternative state.

Once the state diagram for the alternative loop states is defined, it is clear that the steady state phase error for the loop may be obtained by the same method as was used for the first order DPLL in Chapter IV. For the example of Figure 5-2, if the steady state probability of occupancy of alternative state s_i is denoted as a_i , then for state s_1 ,

$$a_1 = p_1 a_{29} + q_1 a_9 \quad (5-4)$$

Similarly for alternative state s_2 ,

$$a_2 = p_1 a_{29} + q_1 a_3 \quad (5-5)$$

Obviously there exists a similar equation for each of the 56 alternative loop states so that there exists a system of homogeneous, linear equations similar to (4-1) which can be written as,

$$[A] \begin{bmatrix} a_1 \\ a_2 \\ \cdot \\ \cdot \\ \cdot \\ a_{56} \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ \cdot \\ \cdot \\ \cdot \\ a_{56} \end{bmatrix} \quad (5-6)$$

where $[A]_{56 \times 56}$ is the stochastic matrix of transition probabilities for the alternative loop states. The nontrivial solution to (5-6) which satisfies the constraint,

$$\sum_{i=1}^{56} a_i = 1 \quad (5-7)$$

gives the steady state probabilities for the alternative loop states. As for the case of the first order DPLL, the steady state probabilities of the reference clock states, S_i , can be determined from the steady state probabilities of the alternative loop states as,

$$S_1 = \sum_{i=1}^{14} a_i \quad (5-8a)$$

$$S_2 = \sum_{i=15}^{28} a_i \quad (5-8b)$$

$$S_3 = \sum_{i=29}^{42} a_i \quad (5-8c)$$

$$S_4 = \sum_{i=43}^{56} a_i \quad (5-8d)$$

The variance of the steady state phase error can then be determined from the distribution of S_i by (4-5). This method of finding the steady state phase error of the loop can obviously be extended to any value of M , N , and L .

The mean time to phase lock for some initial phase offset can also be determined by the method of Chapter IV. If T_{oi} is defined as the mean number of samples required to the first occurrence of phase lock for an initial alternative loop state s_i , then for state s_1 ,

$$T_{o1} = p_3 T_{o29} + p_3 T_{o30} + 1 \quad (5-9)$$

A similar equation can be written for each of the alternative loop states with the exception of the alternative states which define phase lock for the loop. For the example of Figure 5-2, the alternative loop states defining phase lock were found earlier to be s_{19} , s_{25} , s_{32} , and s_{38} . For these states the loop is initially in phase lock so that,

$$T_{o19} = T_{o25} = T_{o32} = T_{o38} = 1$$

Thus there exists a system of non-homogeneous, linear equations similar to (4-10) which can be written as,

$$\begin{array}{c} [T] \end{array} \begin{bmatrix} T_{o1} \\ \cdot \\ \cdot \\ T_{o18} \\ T_{o19} \\ T_{o20} \\ \cdot \\ \cdot \\ \cdot \\ T_{o24} \\ T_{o25} \\ T_{o26} \\ \cdot \\ \cdot \\ \cdot \\ T_{o31} \\ T_{o32} \\ T_{o33} \\ \cdot \\ \cdot \\ \cdot \\ T_{o37} \\ T_{o38} \\ T_{o39} \\ \cdot \\ \cdot \\ \cdot \\ T_{o56} \end{bmatrix} = \begin{bmatrix} 1 \\ \cdot \\ \cdot \\ 1 \\ 0 \\ 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \\ 0 \\ 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \\ 0 \\ 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \\ 0 \\ 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{bmatrix} \quad (5-10)$$

where $[T]$ is 56×56 matrix involving p_i and q_i . As with the steady state phase error, this method of determining the mean number of samples to phase lock can be extended for arbitrary values of M , N , and L .

D. Application of the Markov Chain Model. The use of the Markov chain utilizing the alternative loop states can be generalized for arbitrary values of M , N , and L as has been done via the Fortran programs given in Appendix B. Use of these programs has been outlined in Chapter IV.C. For all cases considered it was assumed that $\Delta_1 = \Delta_2$ and $N = 64$. Further it was found in [9] that the phase error degraded rapidly as L increased, so for all cases considered L was limited to values of 0, 1, or 3.

Using PBDPL2 in Appendix B, the standard deviation of the steady state phase error was determined for $L = 0, 1$, and 3 for various signal-to noise ratios. Figures 5-3, 5-4, and 5-5 give the phase error for values of $M = 1, 2$, and 4, respectively. As can be seen from all three plots there is an approximate 20. db degradation in loop performance for $L = 3$ as opposed to a first order loop ($S = 0$).

Using PBDPL2T given in Appendix B, the transient response of the second order loop was determined for $M = 1, 2$, and 4, $L = 1$ and 3 and signal-to-noise ratios of -20., 0., and 20. db. The results are plotted in Figures 5-6 through 5-11 where for all plots the value of the divide by L counter was zero at the initial phase offset. When the results are compared to those for the first order loop given in Figures 4-6, 4-7, and 4-8, it is seen that for identical values of M and signal-to-noise ratios, that the second order DPLL achieves lock in a lesser expected time for initial phase offsets greater than $\pi/8$ radians. However for smaller initial phase offsets, the first order DPLL actually achieves phase lock in a lesser expected time than

the second order DPLL.

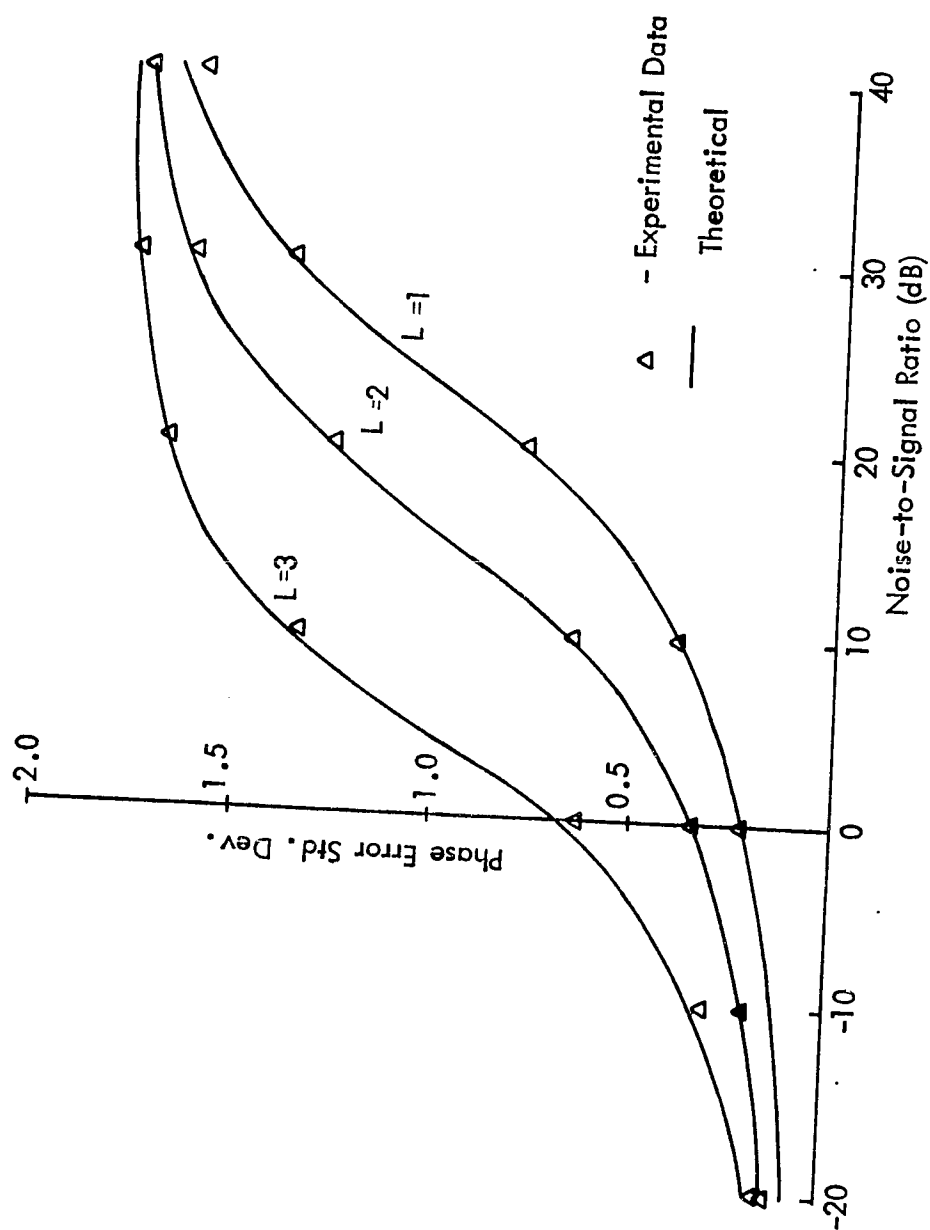


Figure 5-3. Steady State Phase Error $N = 64$, $M = 1$.

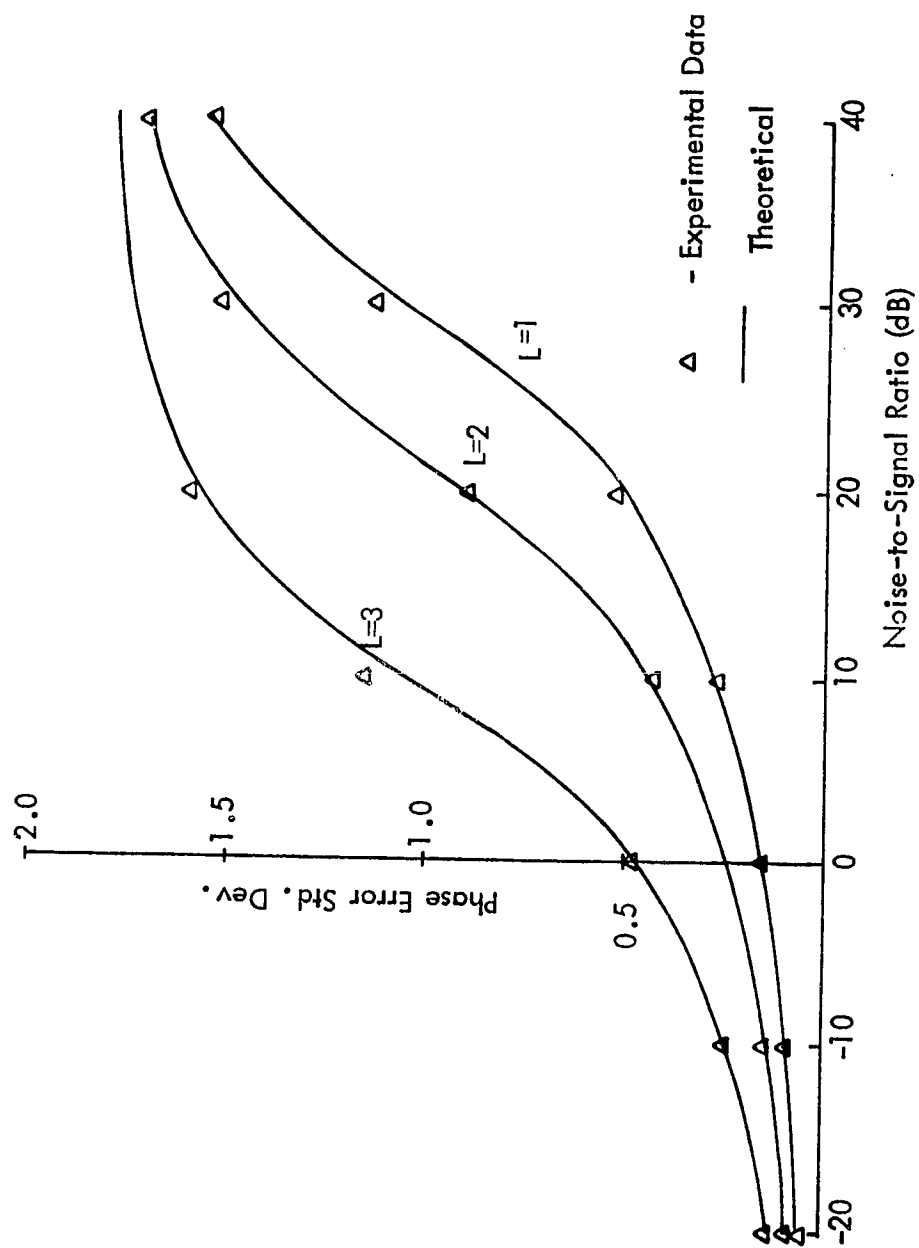


Figure 5-4. Steady State Phase Error $N = 64$, $M = 2$.

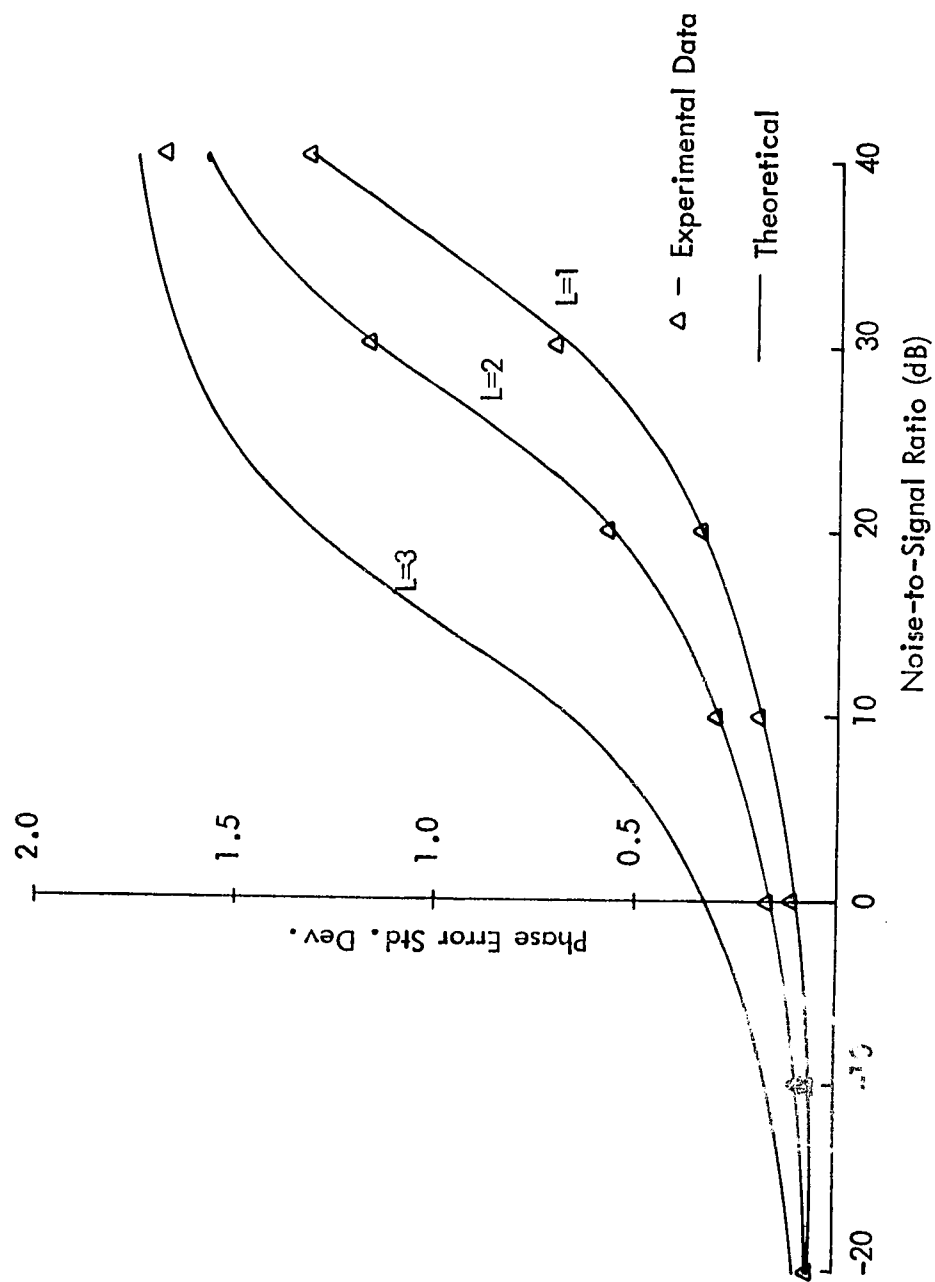


Figure 5-5. Steady State Phase Error $N = 64$, $M = 4$.

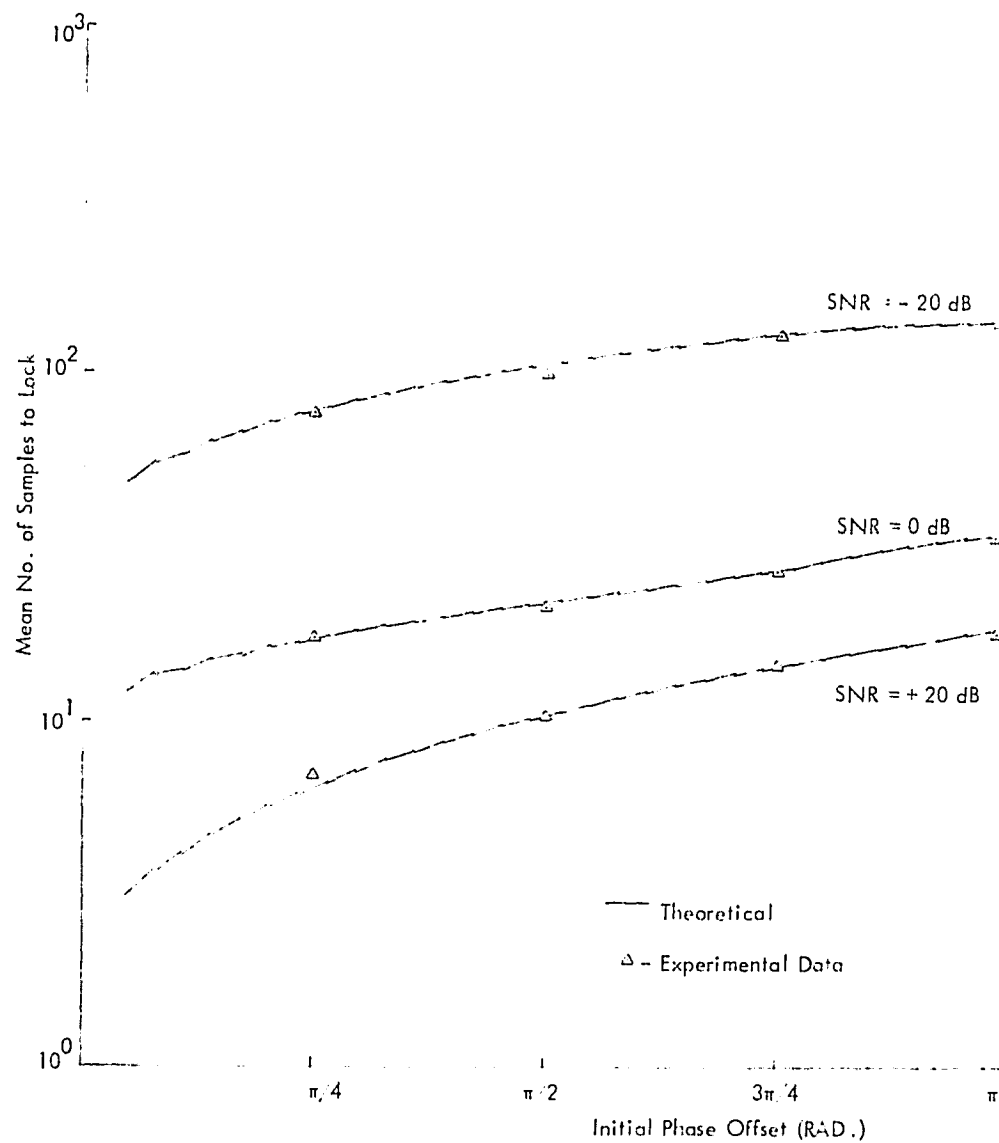


Figure 5-6. Loop Transient Response
 $M=1$, $N=64$, $L=1$.

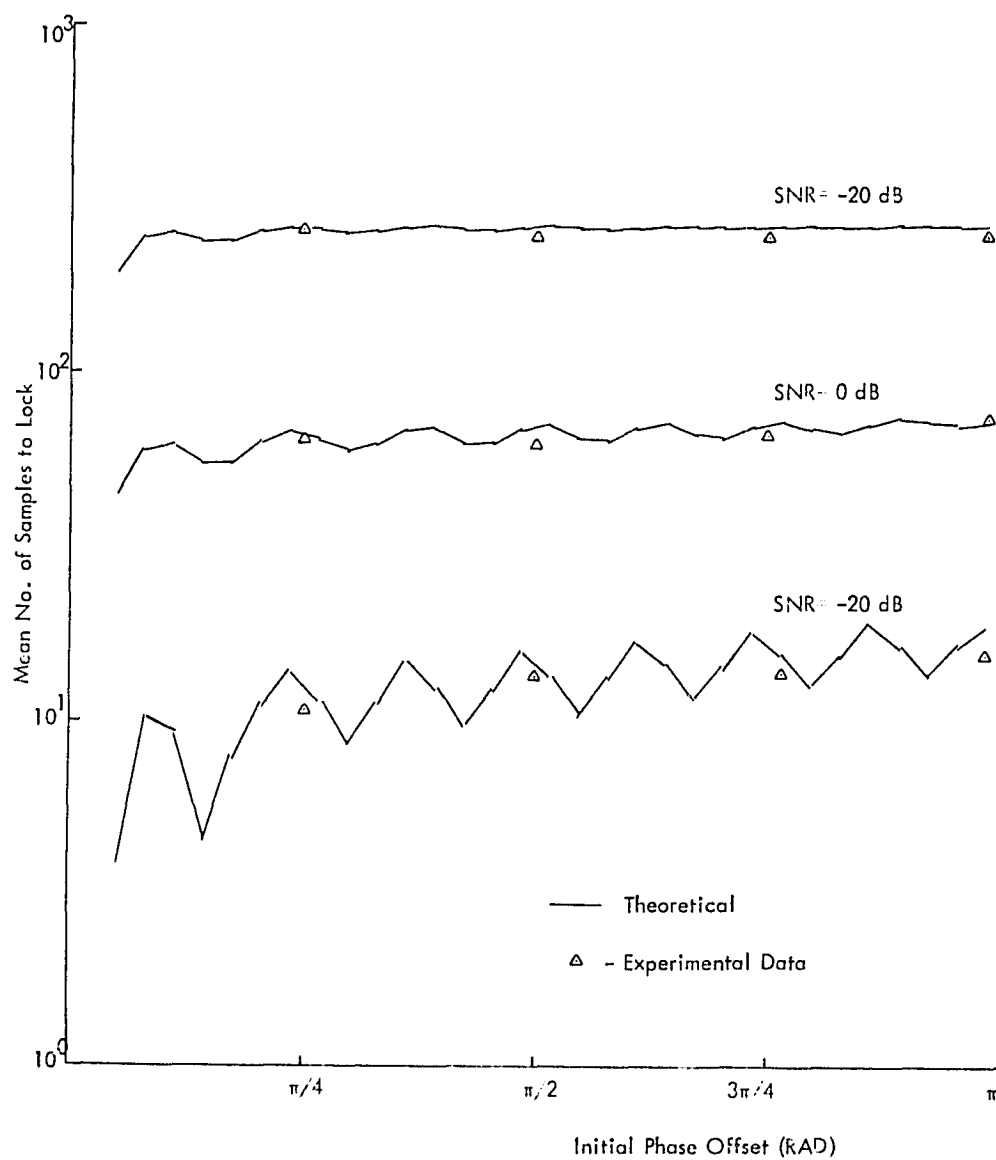


Figure 5-7. Loop Transient Response
 $M=1$, $N=64$, $L=3$.

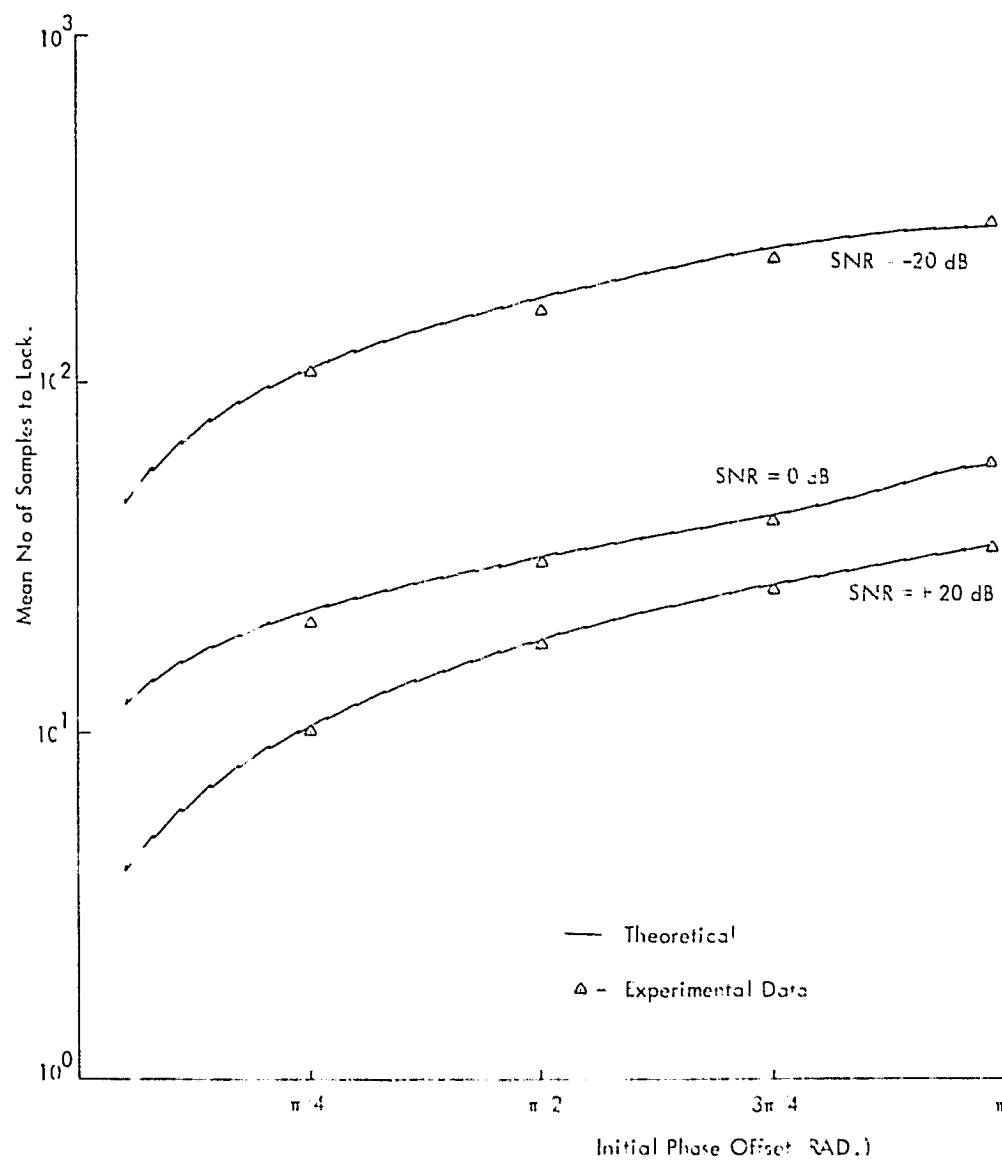


Figure 5-8. Loop Transient Response
 $M=2$, $N=64$, $L=1$.

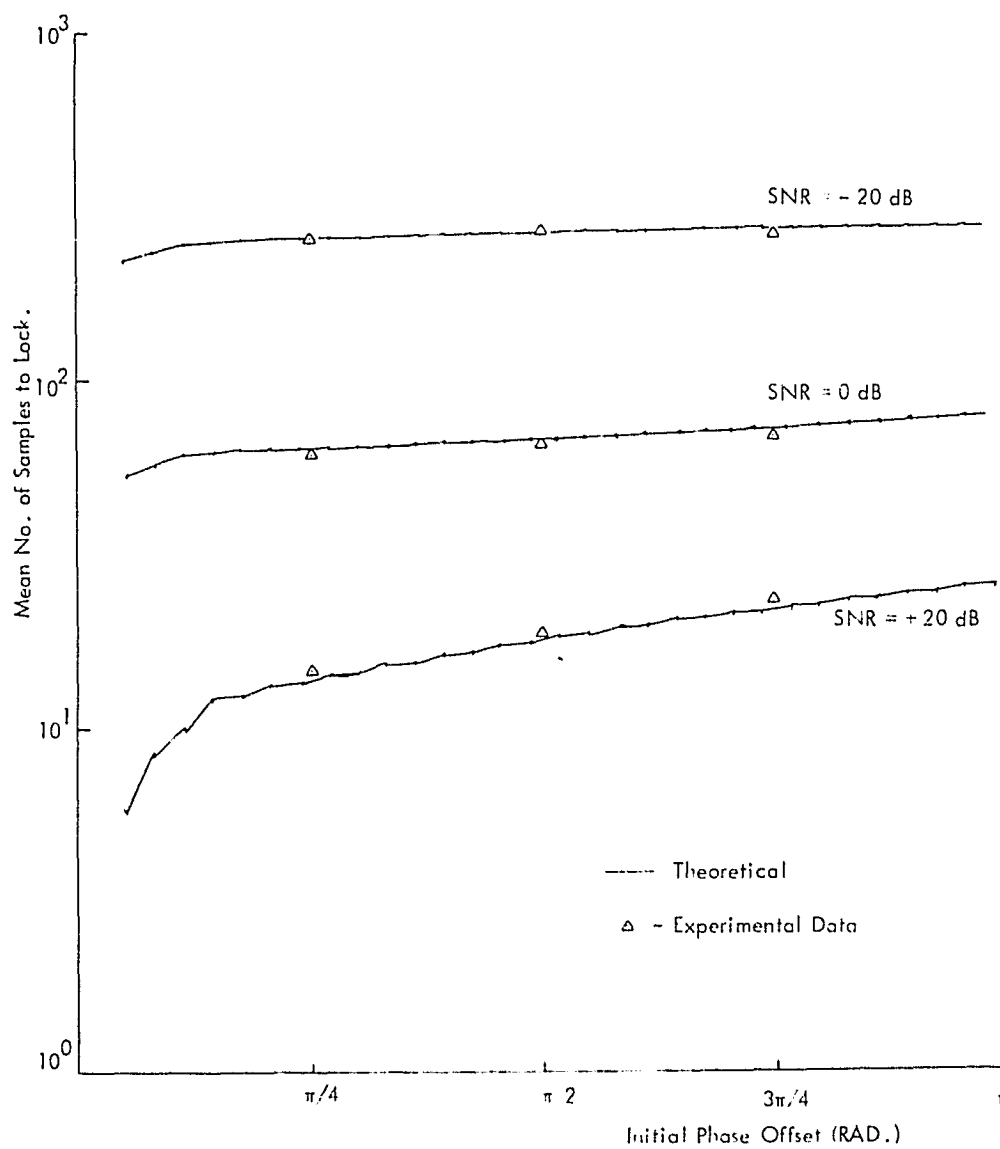


Figure 5-9. Loop Transient Response
 $M=2$, $N=64$, $L=3$.

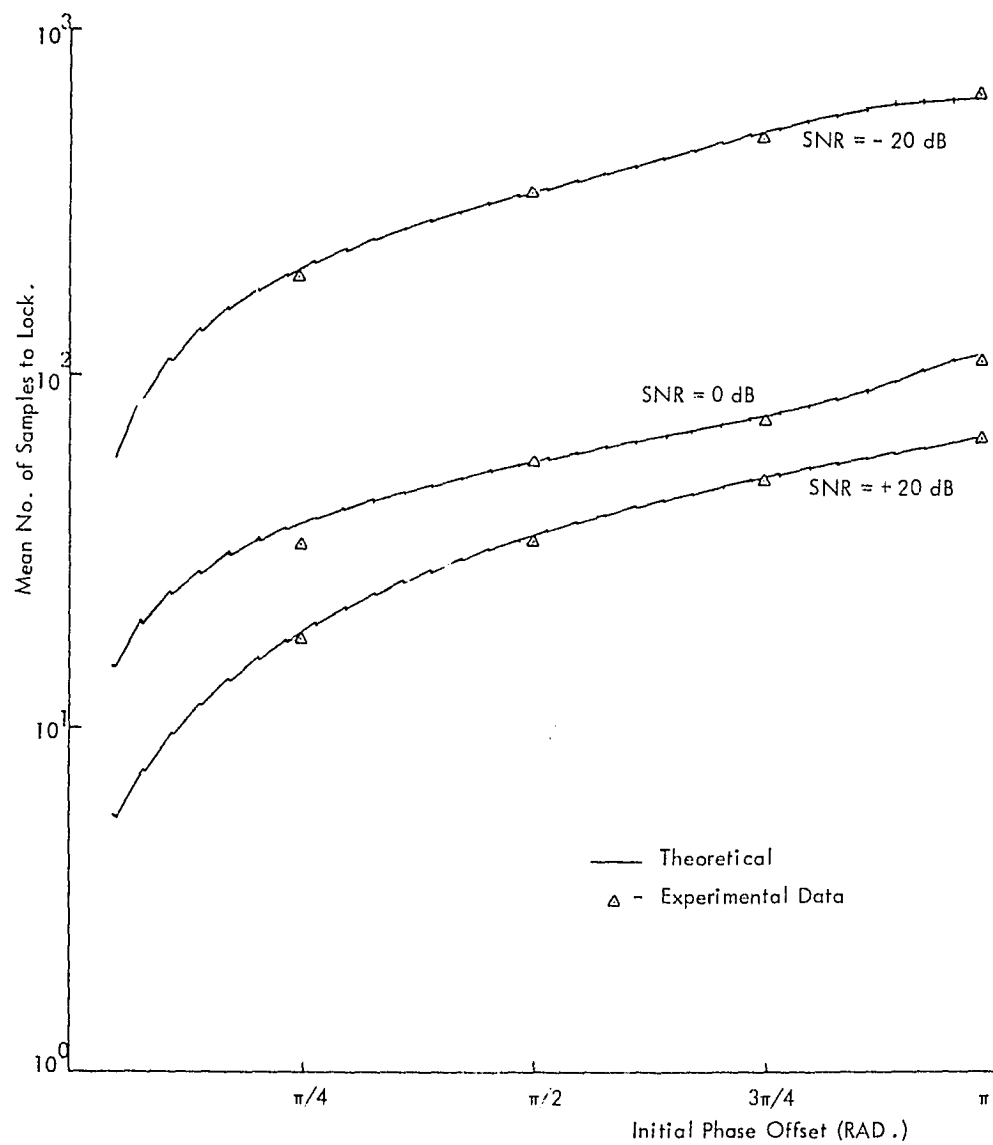


Figure 5-10. Loop Transient Response
 $M=4, N=64, L=1$.

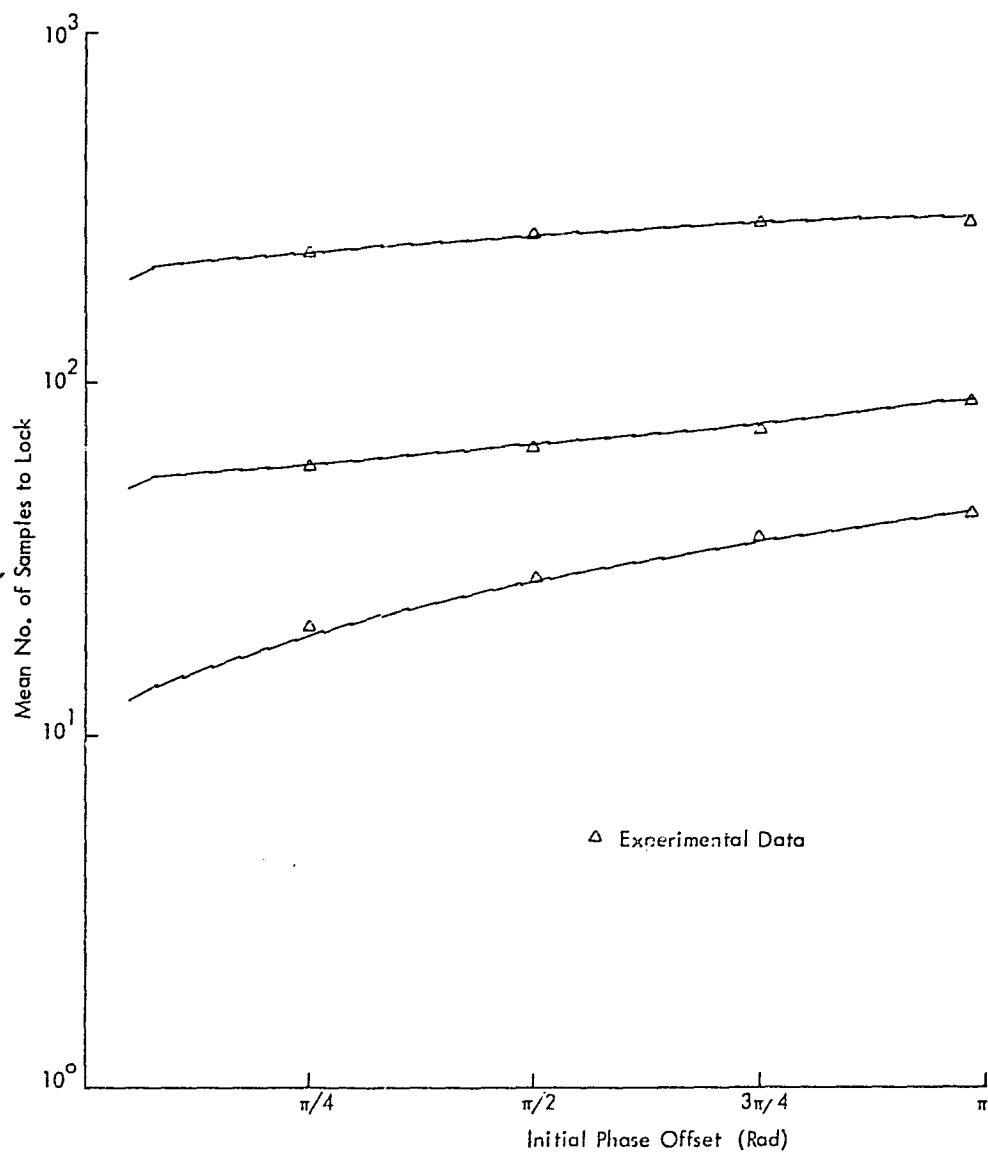


Figure 5-11. Loop Transient Response, $M = 4$, $S = 3$, $N = 64$.

CHAPTER VI

FIRST AND SECOND ORDER DPLL DESIGN

A. Introduction. To experimentally demonstrate the validity of the Markov chain model for the first and second order DPLL's described in Chapters IV and V, a hardware DPLL was designed, constructed and tested for additive noise inputs. The hardware loop was designed so that either first or second order operation was selectable. Also, a primary design objective for the hardware DPLL was to allow for easy alteration of important loop parameters to facilitate testing of the loop under various conditions. Therefore, the design presented is not intended to incorporate minimal components. The general range over which the parameters could be varied was limited to values thought to be useful for an application to an Omega navigation receiver as were the parameters used in Chapters IV and V. It should be emphasized however, that neither the theory nor the general DPLL design presented in this paper are in any way limited to this particular application.

B. DPLL Design and Testing. The block diagram for the DPLL design is shown in Figure 6-1. All register lengths shown in the block diagram indicate the maximum values and during testing the actual register lengths were altered to verify the effects of various loop parameters upon the DPLL's performance. The functional groupings of components in relation to the DPLL given in Figure 5-1 are indicated for comparison.

Referring to Figure 6-1, the phase detector will sample the binary quantized incoming signal at a frequency f_c and then output a count-down signal if the sample is a high level or a count-up signal if the sample is a low level. The count-up or count-down signal is then applied to both the divide-by L saturating counter and the divide-by (MxN) counter. Inhibiting logic is included with the divide-by L counter so that the counter will saturate at selectable values of $\pm(2^i - 1)$,

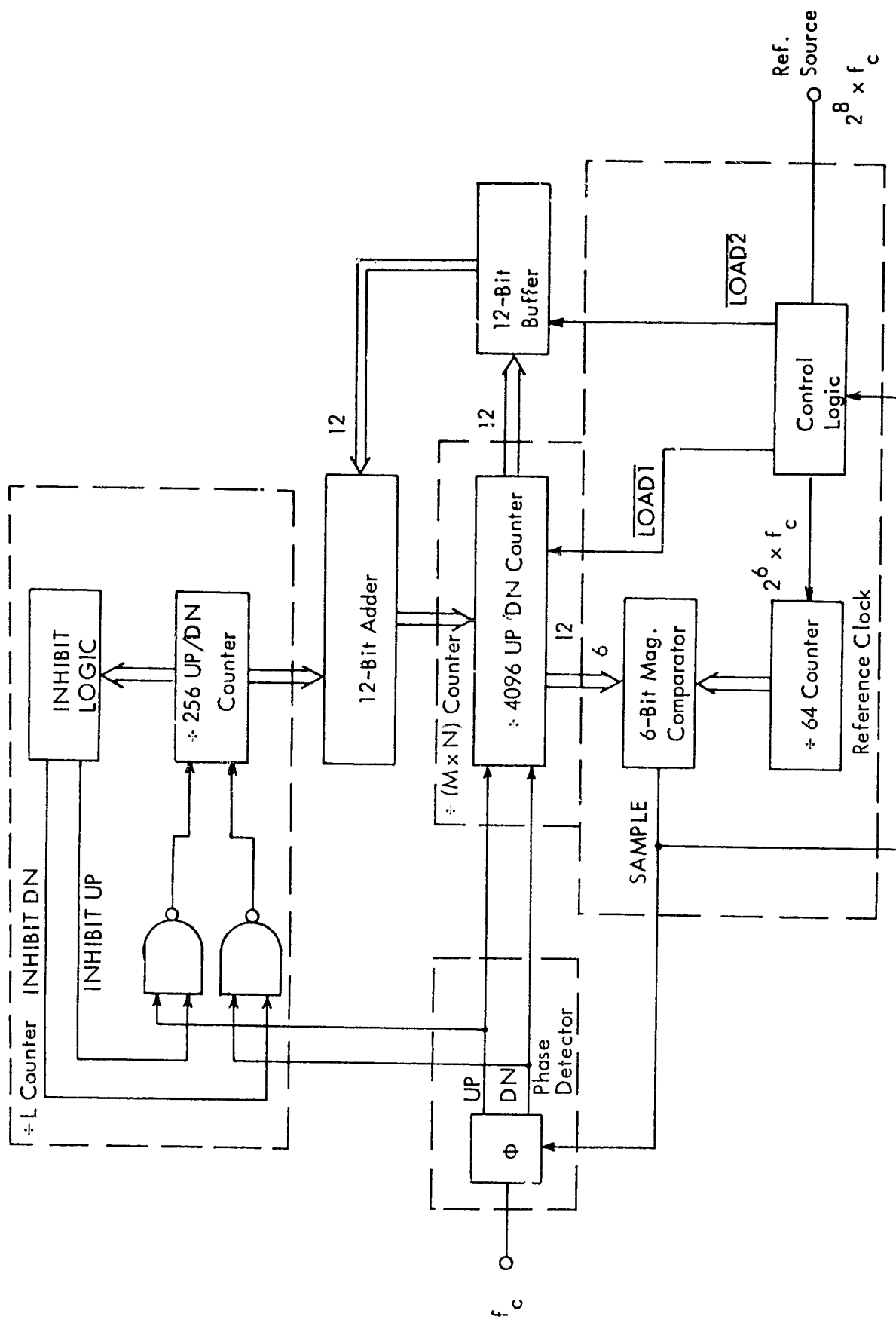


Figure 6-1. Hardware DPLL Block Diagram.

$i = 1, 2, \dots, 7$. The sample command also initiates the control logic so that the new value of the divide-by ($M \times N$) counter (following the count-up or count-down signal) is loaded into the 12-bit buffer by means of the LOAD2 signal. After settling, the output of the 12-bit adder will be the sum of the divide-by L saturating counter and the divide-by ($M \times N$) counter. This value is then loaded into the divide-by ($M \times N$) counter by means of the LOAD1 signal. The value of the N most significant bits of the divide-by ($M \times N$) counter now represents the phase estimate of the DPLL. To establish the variable phase reference clock, this phase estimate is compared to the value of a divide-by N counter being clocked at a rate $N \times f_c$ using a $\log_2 N$ -bit binary magnitude comparator. Upon coincidence of the two input words, the magnitude comparator output takes on value ONE. Note that this pulse output occurs at an f_c rate. Notice also that this loop will operate in the first order mode simply by inhibiting the $\overline{\text{LOAD1}}$ signal to the divide-by ($M \times N$) counter. Detailed schematics for the DPLL may be found in Appendix C.

The primary objective for construction and testing of the hardware DPLL was the verification of the theoretical data obtained from the Markov chain model in Chapters IV and V. As such, it was desired to determine the steady state phase error and the mean time to phase lock for some initial phase offset for the DPLL operating with an additive white gaussian noise input. The determination of these two values can be made using the same basic test configuration shown in Figure 6-2. This test configuration allows the use of identical frequency but phase shifted signals for the DPLL's input and reference source. Also, provision is made for the addition of gaussian noise to the input signal.

Referring to Figure 6-2, the source used for both the DPLL reference source and the DPLL phase shifted input was a Sulzer temperature controlled crystal

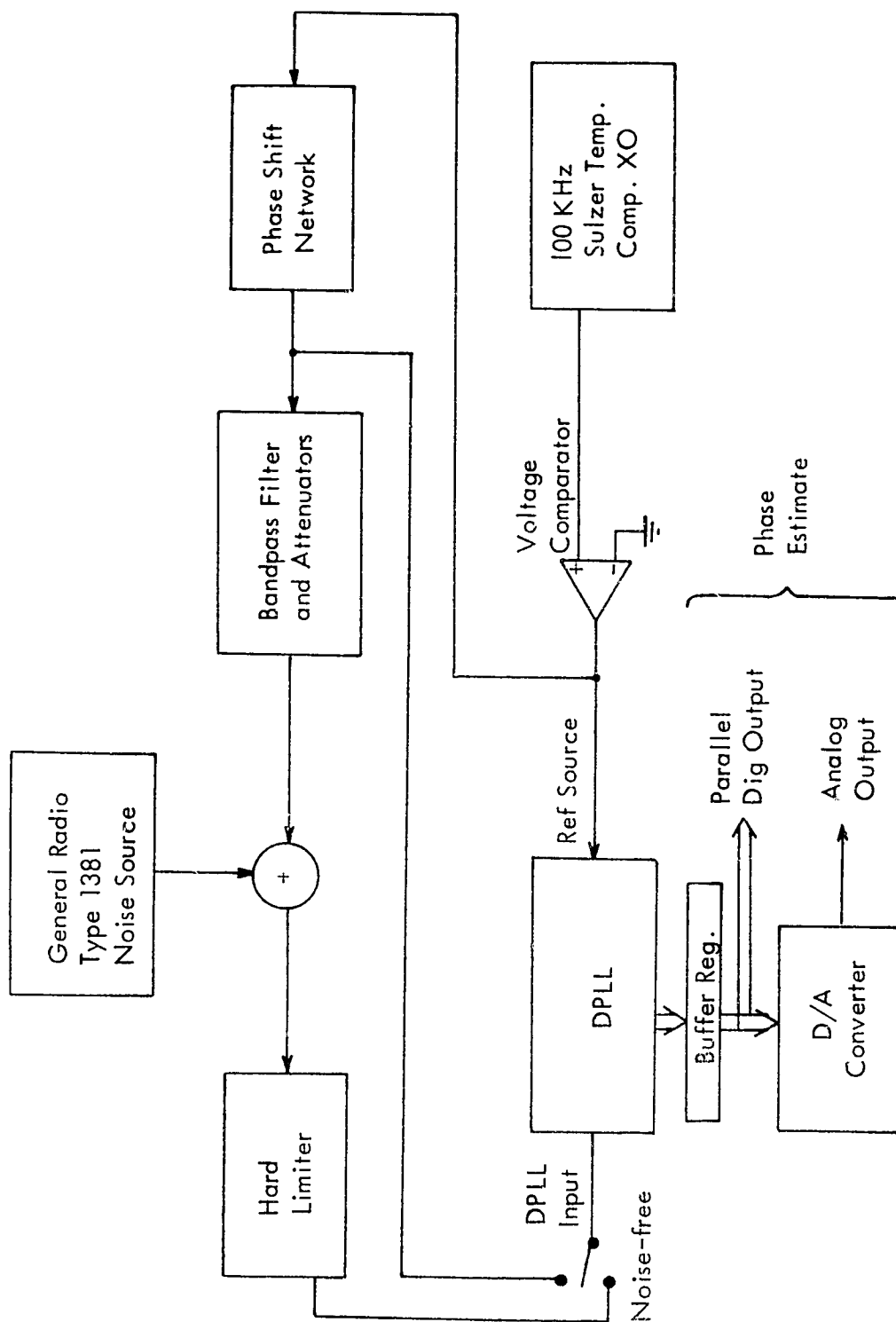


Figure 6-2. DPLL Test Circuit.

oscillator operating at 100 KHz. Since the reference source for the DPLL is required to be $2^8 \times f_c$, this sets the quiescent frequency of the DPLL at approximately 390 Hz. To develop an input signal for the DPLL the 100 KHz hardlimited source is applied to a presettable 8-bit counter. Then by presetting the counter to various values a phase shifted signal at an f_c rate is obtained.

The output of the digital phase shift network was then filtered using a standard state variable active bandpass filter with center frequency of f_c Hz and $Q \approx 100$. Utilization of this particular filter design allows for easy and precise turning of the filter for zero phase shift. The sinusoid output of the bandpass filter was then selectably attenuated using Hewlett-Packard Model 355B attenuators to vary the signal amplitude. Gaussian noise from a General Radio Type 1381 noise source was then added to the attenuated phase shifted sinusoid. The power spectrum for the Type 1381 noise source is flat over a 25 KHz bandwidth and will therefore look to be a white noise source to the DPLL. Noise power was monitored by a HP true-rms voltmeter so that accurate signal-to-noise ratios could be set. The corrupted signal was then hardlimited using a two stage amplifier followed by an LM319 analog comparator. Each amplifier stage consisted of an AD518 op-amp configured for 20. dB of gain at a 40 KHz bandwidth. The output of the LM319 was then used as the noise corrupted hard limited input to the DPLL.

The first test performed on the DPLL involved only the phase detector of the loop. For this, the inputs to the divide-by ($M \times N$) and divide-by L counters of the loop were removed so that the phase of the DPLL's reference clock could not be altered. In this manner, any phase shift between the DPLL's input and reference

clock that was preset by the phase shift network would be maintained. The probability of a count down phase detector output as a function of phase shift and signal-to-noise ratio was then determined simply by counting the number of count down outputs and the total number of phase samples taken. The probability of a count down output is then the ratio of the former number to the latter, and gives an experimental verification of the values of q_i as defined by (4-17a). The experimental results are plotted in Figure 6-3 and 6-4 and show close agreement with the calculated values in all cases.

The DPLL was then reconfigured for proper operation so that steady state phase error test could be performed. For these tests, the phase shift network of Figure 6-2 was set to zero so that in the absence of noise the phase error of the DPLL would be only the quantization error of the loop. The phase state of the DPLL's reference clock was then recorded on a Kennedy incremental digital recorder for each sample of the input signal. The steady state probability for each reference clock state is just the ratio of the number of occurrences of each reference clock state to the total number of reference clock samples taken. The variance of the reference clock phase is then determined from (4-6). The standard deviation of the loop's phase error as determined experimentally has been presented in Chapters IV and V along with the theoretical values predicted by the Markov chain model. Each of the experimental data points represents at least 300,000 samples of the loop phase error and it is obvious that good agreement exists between the experimental and theoretical results.

The final objective for the hardware tests was verification of the predicted

loop transient response. For this, an initial phase offset was applied to the loop via the phase shifting network and the number of samples required to the first occurrence of the phase lock state as defined in Chapters IV and V was recorded. This was done by detecting the states which defined phase lock as given in Chapters IV and V. The detected phase lock state was then used to gate off an event counter that was counting to number of phase detector samples taken following the initial phase offset. Again, the experimental data has been plotted previously along with the theoretical data in Chapters IV and V. Each experimental data point represents at least 500 trials and once again close agreement is seen between the theoretical and experimental results.

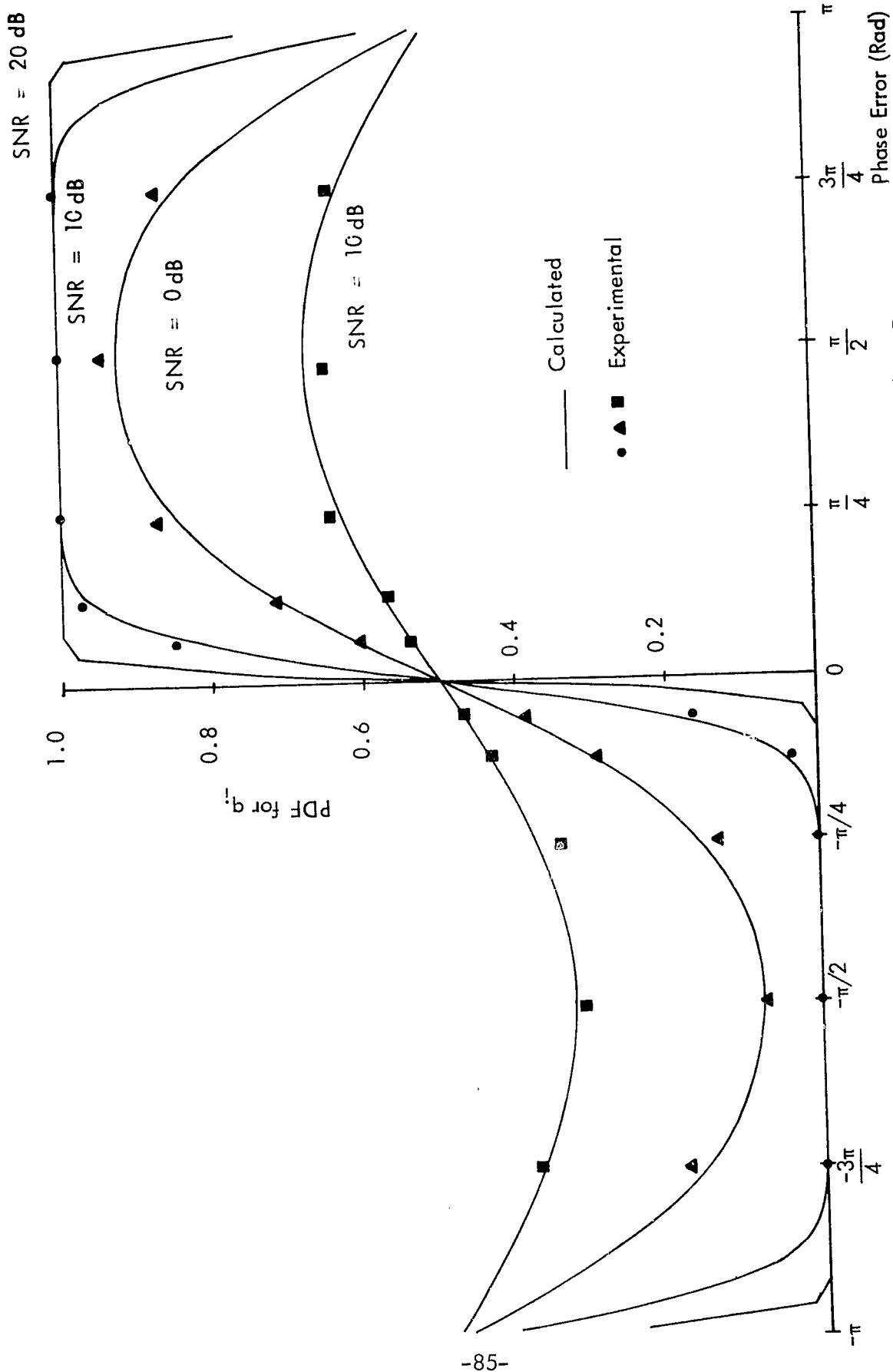


Figure 6-3. Probability Density Function for a Positive Sample at Phase Detector.

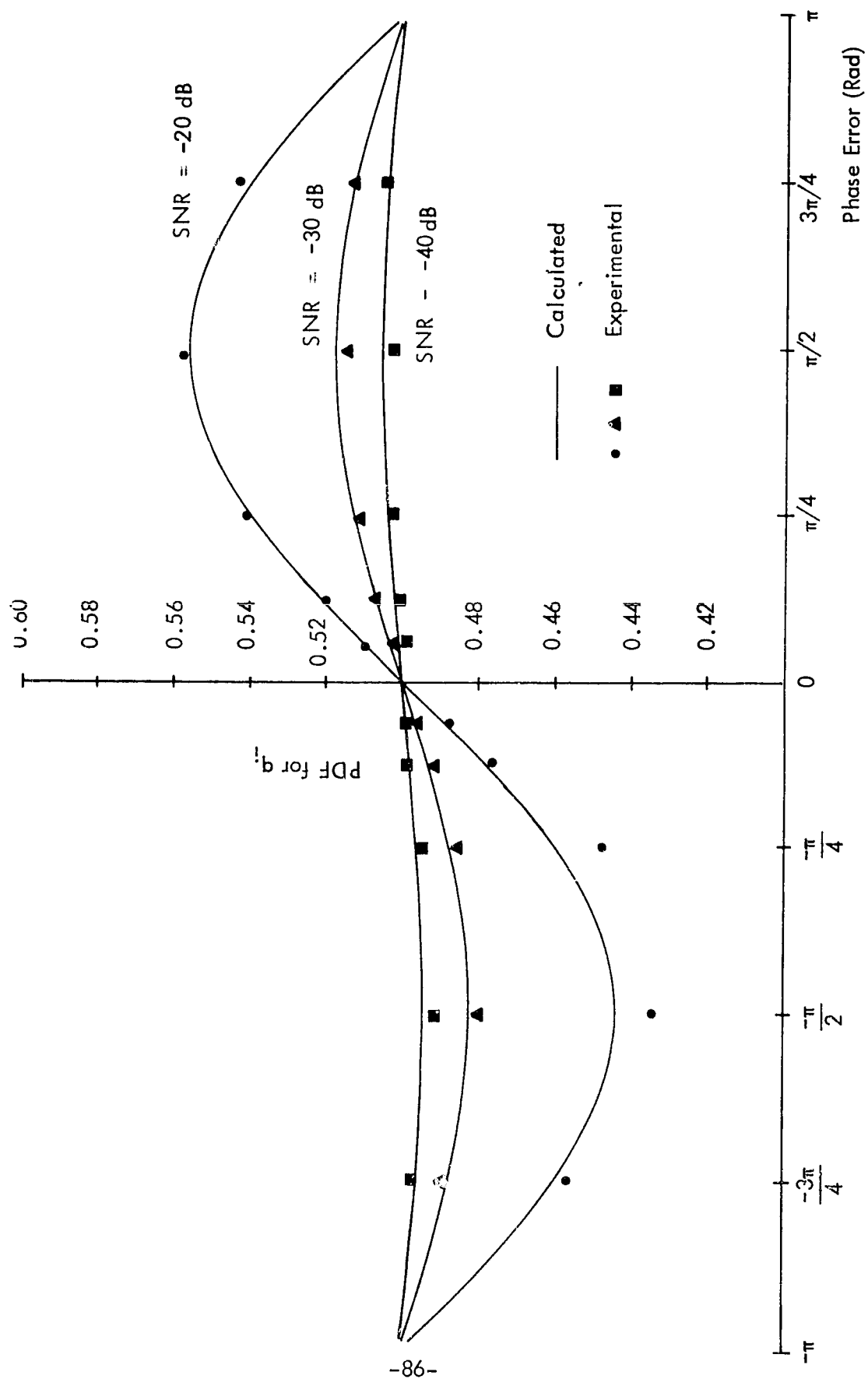


Figure 6-4. Probability Density Function for a Positive Sample at Phase Detector.

CHAPTER VII

CONCLUSIONS AND RECOMMENDATIONS

First and second order all digital phase-locked loops have been analyzed for both ideal and additive gaussian noise inputs. In addition, a hardware DPLL capable of either first or second order operation has been designed and tested for verification of the analytical results. For all cases tested, the experimental data showed close agreement with the analytical results indicating that the Markov chain model for first and second order DPLL's given in Chapters IV and V are valid.

In Chapter III, ideal inputs were considered for both first and second order DPLL's with the objective of classifying the time response of the loops. For both loops it was found that the phase error response was given by a non-linear difference equation for which no direct solution was found. However, partial response characteristics of the phase error was determined for both first and second order DPLL's when the frequency of the input signal is identical to that of the loop's quiescent frequency. Also, expressions for the frequency range over which phase lock can be achieved for a first order DPLL and for the frequency range for which a second order DPLL can achieve phase lock in minimum time were derived. In both cases it was found that the frequency range was directly dependent upon the number of distinct phase states of the reference clock. As would be expected, it was found that the frequency range for which a second order DPLL will achieve phase lock, even with the constraint of minimum time to lock, is significantly greater than the frequency range over which a first order DPLL will achieve phase lock.

Specific first and second order DPLL's were also analyzed for stochastic inputs by means of a Markov chain model in Chapters IV and V, respectively. From this

Markov chain model, the steady state phase error and mean transient response were determined. The loop configurations used for the noise analysis were specifically chosen to both match the general loop model given in Chapter III and to be realized in hardware by standard binary logic families. For both first and second order loops it was found that the usual tradeoff between steady state error and transient response existed. That is, the steady state error can be decreased only with the cost of a longer transient response and the transient response can be decreased only with an increase in steady state error.

For the data presented in Chapters IV and V several specific points are worth noting. First, in comparing the transient response of the first and second order DPLL it is found that the first order DPLL will achieve phase lock in less expected time than the second order DPLL for initial phase offsets less than approximately $\pi/8$. Also, for a second order DPLL, the steady state phase error degrades rapidly as the signal-to-noise ratio decreases below 0.0 dB. Thus for an application such as an Omega receiver, if the initial phase error as the received signal is gated on is expected to be small then a first order DPLL will perform in a superior manner over a second order DPLL. However, if the signal-to-noise ratio can be expected to be in excess of 0.0 dB and the initial phase error is unknown then the second order DPLL will give superior performance.

Completion of the goals of this paper also points to areas in which further research would be of use. In particular, for the case of an ideal input, it would be of great utility if the determination of the phase error response characteristics were extended to include the case of frequency offsets. This extension would then allow the determination of time required to lock for an initial frequency offset without the necessity of

performing a simulation. For the case of stochastic inputs, further research on several points is recommended. First, the concepts of Chapters IV and V should be extended to determine pertinent statistics on loop cycle-slippage. Secondly, it would be useful to develop a comparison of the DPLL's transient response to the standard loop bandwidth used to characterize APLL's. Finally, because of the transient response/steady state error tradeoff previously mentioned research into adaptive loops could provide an optimum relationship between these two parameters.

VIII. ACKNOWLEDGMENTS

The research presented in this dissertation has been partially sponsored by the National Aeronautics and Space Administration at Langley Research Center as part of the Tri-University program. This work was performed at the Avionics Engineering Center at Ohio University, Athens, Ohio.

The author would like to acknowledge the assistance of Dr. J. E. Essman, Assistant Dean, College of Engineering and Technology and Dr. R. H. McFarland, Director, Avionics Engineering Center. In addition, the assistance of Dr. R. W. Lilley, Assistant Director, Avionics Engineering Center has been invaluable throughout the course of this research.

IX. REFERENCES

- [1] Viterbi, A.J., Principles of Coherent Communications, McGraw-Hill Book Company, New York, NY, 1966, Chapters 2 and 3.
- [2] Lindsey, W.C. and Simon, M.K., Telecommunications Systems Engineering, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1973.
- [3] Reddy, C.P., and Gupta, S.C., "A Class of All Digital Phase-Locked Loops: Modeling and Analysis", IEEE Trans. on Industrial Electronics and Control Instrumentation, vol IECI-20, pp.239-251, November, 1973.
- [4] Weinburg, A. and Liu, B., "Discrete Time Analysis of Nonuniform Sampling First- and Second-Order Digital Phase-Locked Loops", IEEE Trans. on Communications, vol COM-22, pp. 123-137, February, 1974.
- [5] Cessna, J.R. and Levy, D.M., "Phase Noise and Transient Response for a Binary Quantized Digital Phase-Locked Loop in White Gaussian Noise", IEEE Trans. on Communications, vol COM-20, pp.94-103, April, 1972.
- [6] Holmes, J.K., "Performance of a First Order Transition Sampling Digital Phase-Locked Loop Using Random Walk Models", IEEE Trans. on Communications, vol COM-20, pp.119-131, April, 1972.
- [7] Chamberlin, K.A., "Binary Phase-Locked Loops for Omega Receivers", Proceedings of Second Omega Symposium, The Institute of Navigation, Washington, D.C., pp. 154-159, November, 1974.
- [8] Feller, W., An Introduction to Probability Theory and Its Applications, Volume 1, John Wiley and Sons, New York, NY, 1950.
- [9] Blasche, P.R., "Simulation of Digital Phase-Locked Loops", NASA Technical Memorandum No. 49, Avionics Engineering Center, Ohio University, Athens, Ohio, May, 1977.
- [10] Gill, G.S., and Gupta, S.C., "On Higher Order Discrete Phase-Locked Loops", IEEE Trans. on Aerospace and Electronic Systems, vol AES-8, pp. 615-623, September, 1972.
- [11] Gupta, S.C., "Phase-Locked Loops", Proceedings of the IEEE, vol 63, pp. 291-305, February, 1975.
- [12] Lee, W.H., Harrington, E.V., and Cox, D.B., "A New Integrated Circuit Digital Phase-Locked Loop, Report P-148, The Charles Stark Draper Laboratory, Inc., March, 1975.

- [13] Holmes, J.K., and Tegenelia, C.R., "A Second-Order All-Digital Phase-Locked Loop", IEEE Trans. on Communications, vol COM-22, pp. 62-68, January, 1974.
- [14] Yamamoto, H., and Mori, S., "Performance of a Binary Quantized All Digital Phase-Locked Loop with a New Class of Sequential Filter", IEEE Trans. on Communications, vol COM-26, pp. 35-45, January, 1978.
- [15] Ransom, J.J., and Gupta, S.C., "Performance of a Finite Phase State Bit-Synchronization Loop With and Without Sequential Filters", IEEE Trans. on Communications, vol COM-23, pp. 1198-1206, November, 1975.

X. APPENDICES

A. Markov Chains. A Markov chain can be characterized by a system containing a number of distinguishable states (finite or infinite) for which transition to a new state denoted s_j from any present state s_i depends solely on the present state s_i . For the application of interest in this paper, assume that the number of system states is N , finite. For each state to state transition, define p_{ij} as the conditional probability of a transition to state s_j given that the present state is s_i . Then for a_k defined as the probability that the system is initially in state s_k , the conditional probability that the system has traversed through states $s_{i1}, s_{i2}, \dots, s_{in-1}$ to the present state s_{in} is given by,

$$P_r [s_{in} | s_{i1}, s_{i2}, \dots, s_{in-1}] = a_{i1} p_{i1, i2} p_{i2, i3} \dots p_{in-1, in} \quad (A-1)$$

For each system state s_i , $i = 1, 2, \dots, N$ a vector of transition probabilities, P_i may be written as,

$$P_i = (p_{i1} \ p_{i2} \ \dots \ p_{iN}) \quad i = 1, 2, \dots, N \quad (A-2)$$

Note that the N transition probability vectors must satisfy the requirements,

$$p_{ij} \geq 0 \quad i, j = 1, 2, \dots, N \quad (A-3)$$

and

$$\sum_{j=1}^N p_{ij} = 1.0 \quad i = 1, 2, \dots, N \quad (A-4)$$

and are called stochastic vectors. The N vectors may be arranged in a stochastic matrix of size $N \times N$ to give the matrix of transition probabilities, $[P]$,

$$[P] = \begin{bmatrix} p_{11} & p_{12} & \cdot & \cdot & \cdot & p_{1N} \\ p_{21} & p_{22} & \cdot & \cdot & \cdot & p_{2N} \\ \cdot & \cdot & & & & \cdot \\ \cdot & \cdot & & & & \cdot \\ \cdot & \cdot & & & & \cdot \\ p_{N1} & p_{N2} & \cdot & \cdot & \cdot & p_{NN} \end{bmatrix} \quad (A-5)$$

This matrix then defines all of the state to state transitions for the system and together with the initial state distribution $a_i, i = 1, 2, \dots, N$ for states s_1, s_2, \dots, s_N completely defines the Markov chain for states s_1, s_2, \dots, s_N .

The matrix of (A-6) defines the probabilities for state to state transitions for a single step. Next consider a transition from state s_i to state s_j that occurs in exactly n steps. Denote $p_{ij}^{(n)}$ as the probability of observing the system in state s_j at time $r + n$ given that at time r it was observed in state s_i . Obviously there will exist a number of different paths for which the transition from s_i to s_j will occur in exactly n steps and $p_{ij}^{(n)}$ will be the sum of the probabilities for all of the possible paths. For example,

$$p_{ij}^{(1)} = p_{ij} \quad (A-6)$$

and

$$p_{ij}^{(2)} = \sum_{k=1}^N p_{ik} p_{kj} \quad (A-7)$$

It can be shown by induction that $p_{ij}^{(n)}$ is given by the recursive relationship,

$$p_{ij}^{(n)} = \sum_{k=1}^N p_{ik} p_{kj}^{(n-1)} \quad (A-8)$$

If the value $p_{ij}^{(n)}$ are arranged as elements of a matrix denoted $[P]^n$ then it is obvious from (A-8) that standard matrix multiplication is applicable so that

$$[P]^n = [P][P]^{n-1} \quad (A-9)$$

Recall that a_i was defined as the probability of observing the system in state s_i at time zero. Then the unconditional probability of observing the system in state s_i after n steps is

$$a_i^{(n)} = \sum_{j=1}^N a_j p_{ji}^{(n)} \quad (A-10)$$

Further, the distribution of $a_i^{(n)}$ will tend to be independent of the initial distribution a_i , $i = 1, 2, \dots, N$ if $p_{ji}^{(n)}$ is independent of i which will be the case if $[P]^n$ converges to a matrix of identical rows. If $[P]^n$ does converge, then as $n \rightarrow \infty$ the distribution $a_i^{(n)}$ is the steady state distribution for the system.

It is now necessary to show that such a steady state distribution exists.

First, define the states of an aperiodic Markov chain as being transient if the probability of reoccurrence of that state is less than one for infinite time so that

$$\sum_{n=1}^{\infty} p_{ii}^{(n)} < \infty \quad (A-11)$$

and define the state as being null if the probability of reoccurrence of that state is one but the mean time to reoccurrence is infinite. It has been proven by Feller [8], Chapter XV.6 that for an aperiodic Markov chain that all states are either transient or null or all states are ergodic. If all states are ergodic then

$$\lim_{n \rightarrow \infty} p_{kk}^{(n)} = U_k > 0 \quad (A-12)$$

where $1/U_k$ is the mean reoccurrence time of state s_k and U_k , $k = 1, 2, \dots, N$

is the unique steady state distribution for the system.

A Markov chain can also be specified for which each state has one or more possible vectors of transition probabilities. For each state s_i , the possible vectors are called alternative vectors and transition from the present state to some new state is governed by one and only one of the alternative vectors associated with the present state. In this case, for each state s_i there exists K_i alternative vectors,

$$k_{p_i} = (k_{p_{i1}} \ k_{p_{i2}} \ \dots \ k_{p_{iN}}) \quad i = 1, 2, \dots, N \quad (A-13)$$

$$k = 1, 2, \dots, K_i$$

where p_{ij} is the probability that the system will make a transition to state s_j given that the present state is s_i and k^{th} alternative for s_i is used. As before, each of the alternative vectors of transition probabilities are stochastic vectors and must satisfy the conditions,

$$k_{p_{ij}} \geq 0 \quad i, j = 1, 2, \dots, N \text{ and } k = 1, 2, \dots, K_i \quad (A-14)$$

and

$$\sum_{j=1}^N k_{p_{ij}} = 1.0 \quad i = 1, 2, \dots, N \text{ and } k = 1, 2, \dots, K_i \quad (A-15)$$

The vectors may be combined to give the $K \times N$ stochastic matrix, \underline{P} ,

$$\begin{aligned}
 \underline{P} = & \begin{bmatrix}
 {}^1p_{11} & {}^1p_{12} & \cdot & \cdot & \cdot & {}^1p_{1N} \\
 {}^2p_{11} & {}^2p_{12} & \cdot & \cdot & \cdot & {}^2p_{1N} \\
 \cdot & \cdot & & & & \cdot \\
 \cdot & \cdot & & & & \cdot \\
 \cdot & \cdot & & & & \cdot \\
 {}^{k_1}p_{11} & {}^{k_1}p_{12} & \cdot & \cdot & \cdot & {}^{k_1}p_{1N} \\
 {}^1p_{21} & {}^1p_{22} & \cdot & \cdot & \cdot & {}^1p_{2N} \\
 {}^2p_{21} & {}^2p_{22} & \cdot & \cdot & \cdot & {}^2p_{2N} \\
 \cdot & \cdot & & & & \cdot \\
 \cdot & \cdot & & & & \cdot \\
 \cdot & \cdot & & & & \cdot \\
 {}^{k_2}p_{21} & {}^{k_2}p_{22} & \cdot & \cdot & \cdot & {}^{k_2}p_{2N} \\
 \cdot & \cdot & & & & \cdot \\
 \cdot & \cdot & & & & \cdot \\
 \cdot & \cdot & & & & \cdot \\
 {}^{k_N}p_{N1} & {}^{k_N}p_{N2} & \cdot & \cdot & \cdot & {}^{k_N}p_{NN}
 \end{bmatrix}
 \end{aligned}
 \tag{A-16}$$

B. Program Listings. The following three fortran programs were written to solve for the DPLL's performance characteristics as described by the Markov model given in Chapters IV and V.

1. PBSTGEN. Program PBSTGEN is used to establish the systems of equations associated with (4-1)/(5-2) and (4-9)/(5-10). For these systems of equations it is seen that only two elements of each row are non-zero. Therefore, instead of storing the entire coefficient matrix for these systems of equations, PBSTGEN generates two vectors whose lengths are twice the dimension of the coefficient matrices for each system. The first vector, PNTRI for (4-1)/(5-2) and T1 for (4-9)/(5-10), contains the alternative loop state number for each equation (i.e. the possible state to state transitions for each equation) while the second vector, PNTR2 for (4-1)/(5-2) and T2 for (4-9)/(5-10), contain a pointer for the probability coefficient associated with each state to state transition. The actual value of these probability coefficients will be calculated in the following programs.

2. PBDPL2. Program PBDPL2 uses the vectors PNTRI and PNTR2 generated in PBSTGEN to solve for the steady state phase error of the DPLL. Solution is performed by means of Jacobi's iterative technique until the steady state probability for all alternative loop states converges a difference of less than 10^{-8} for successive iterations. If convergence is not achieved, a maximum of 10,000 iterations are performed. The iterative solution is performed for signal-to-noise ratios from -40. dB to 20. dB in increments of 5. dB and the output consists of the steady state probability of each reference clock state and the standard deviation of the phase error.

3. PBDPL2T. Program PBDPL2T uses the vectors T1 and T2 generated by PBSTGEN to solve for the mean time to phase lock for an initial phase offset. Solution is performed by Jacobi's iterative technique until the result converges to a difference of less than 10^{-5} for successive iterations. Again, a maximum of 10,000 iterations will be performed. The solution is performed for signal-to-noise ratios from -40. dB to 20. dB in increments of 5. dB. The output consists of the mean time to lock for initial phase offsets from $-\pi$ to π where the value of the divide-by L counter is initially zero.

```

C*****P8S00010
C                                     P8S00020
C      CALCULATION OF STATE TRANSFER VECTORS FOR DPLL          P8S00030
C      MARKOV CHAIN MODEL. THIS PROGRAM TO BE USED IN         P8S00040
C      CONJUNCTION WITH PBDPL2 AND PBDPL2T.                   P8S00050
C                                                                P8S00060
C      INPUT READ ON DEVICE 5                                   P8S00070
C      M=VALUE OF DIVIDE-BY M COUNTER                          P8S00080
C      N=VALUE OF DIVIDE-BY N COUNTER                          P8S00090
C      L=VALUE OF DIVIDE-BY L COUNTER                          P8S00100
C                                                                P8S00110
C      OUTPUT-STEADY STATE PHASE ERROR DATA (WRITE ON DEVICE 6) P8S00120
C      M, N, AND L AS DEFINED FOR INPUT                        P8S00130
C      PTR1= VECTOR CONTAINING THE STATE TO STATE TRANSITIONS P8S00140
C      POSSIBLE                                                P8S00150
C      PTR2= VECTOR CONTAINING THE PROBABILITIES FOR THE STATE P8S00160
C      TO STATE TRANSITIONS.                                  P8S00170
C                                                                P8S00180
C      OUTPUT-MEAN TRANSIENT RESPONSE DATA (WRITE ON DEVICE 9) P8S00190
C      M, N, AND L AS DEFINED FOR INPUT                        P8S00200
C      T1=VECTOR CONTAINING THE STATE TO STATE TRANSITIONS    P8S00210
C      POSSIBLE                                                P8S00220
C      T2=VECTOR CONTAINING THE PROBABILITIES FOR THE STATE    P8S00230
C      TO STATE TRANSITIONS.                                  P8S00240
C                                                                P8S00250
C*****P8S00260
C      INTEGER PTR1(7168),PTR2(7168),IN(7,512),ALT,ALT2       P8S00270
C      INTEGER T1(7168),T2(7168)                               P8S00280
C                                                                P8S00290
C      SET DPLL PARAMETERS                                     P8S00300
C                                                                P8S00310
C      READ(5,3) M,N,L                                         P8S00320
C      3 FORMAT(3I5)                                           P8S00330
C      WRITE(8,2)N,M,L                                         P8S00340
C      WRITE(9,2)N,M,L                                         P8S00350
C      2 FORMAT(1X,'N=',13,5X,'M=',13,5X,'L=',13)           P8S00360
C      ALT=2*L+1                                                P8S00370
C      MN=M*N                                                  P8S00380
C      DO 10 I=1,ALT                                           P8S00390
C      K=1                                                       P8S00400
C      DO 20 J=1,MN                                             P8S00410
C      IN(I,J)=K                                                P8S00420
C      20 K=K+ALT                                               P8S00430
C      10 CONTINUE                                             P8S00440
C      L=1                                                       P8S00450
C      KALT=(ALT+1.)/2.                                         P8S00460
C      ALT2=ALT/2                                               P8S00470
C      K=1                                                       P8S00480
C      DO 40 J=1,MN                                             P8S00490
C      DO 30 I=1,ALT                                           P8S00500
C      RI=1                                                       P8S00510
C      IF(RI-KALT) 50,60,70                                     P8S00520
C      50 IF(I.EQ.1) GO TO 51                                    P8S00530
C      IK=I-1                                                    P8S00540
C      JK=J+ALT2+1                                              P8S00550

```

IK1=I+1	PBS00560
JK1=J+ALT2-1-1	PBS00570
GO TO 80	PBS00580
51 IK=I	PBS00590
JK=J+ALT2+1	PBS00600
IK1=I+1	PBS00610
JK1=J+ALT2-2	PBS00620
GO TO 80	PBS00630
60 IF(ALT.EQ.1) GO TO 51	PBS00640
IK=I-1	PBS00650
JK=J+2	PBS00660
IK1=I+1	PBS00670
JK1=J-2	PBS00680
GO TO 80	PBS00690
61 IK=J	PBS00700
JK=J+1	PBS00710
IK1=I	PBS00720
JK1=J-1	PBS00730
GO TO 80	PBS00740
70 IF(I.EQ.ALT) GO TO 71	PBS00750
IK=I-1	PBS00760
JK=J-1+ALT2+3	PBS00770
IK1=I+1	PBS00780
JK1=J-1+ALT2-1	PBS00790
GO TO 80	PBS00800
71 IK=I-1	PBS00810
JK=J-ALT2+2	PBS00820
IK1=I	PBS00830
JK1=J-ALT2-1	PBS00840
80 IF(JK.GT.MN) JK=JK-MN	PBS00850
IF(JK1.GT.MN) JK1=JK1-MN	PBS00860
IF(JK.LT.1) JK=JK+MN	PBS00870
IF(JK1.LT.1) JK1=JK1+MN	PBS00880
PTRI(K)=IN(IK,JK)	PBS00890
PTRI(K+1)=IN(IK1,JK1)	PBS00900
K=K+2	PBS00910
30 CONTINUE	PBS00920
IF(MOD(J,M).EQ.0) L=L+1	PBS00930
40 CONTINUE	PBS00940
NST=K-1	PBS00950
K=1	PBS00960
LC=-1	PBS00970
LC1=0	PBS00980
93 DO 90 I=1,NST	PBS00990
IF(MOD(I,2).NE.0) GO TO 94	PBS01000
IF(LC1.NE.1) GO TO 92	PBS01010
94 IF(PTRI(I).NE.(K+1)/2) GO TO 92	PBS01020
T1(K)=(I+1)/2	PBS01030
PTRI(K)=(((T1(K)+M-1)/M)+ALT-1)/ALT	PBS01040
K=K+1	PBS01050
LC=LC+1	PBS01060
LC1=1	PBS01070
92 IF(LC.NE.1) GO TO 90	PBS01080
LC=-1	PBS01090
LC1=0	PBS01100

GO TO 93	PBS01110
90 CONTINUE	PBS01120
IF(K.GT.NST) GO TO 96	PBS01130
LC1=1	PBS01140
GO TO 93	PBS01150
96 DO 100 I=2,NST,2	PBS01160
100 PTR2(I)=PTR2(I)+N	PBS01170
IF(ALL.EQ.1) GO TO 101	PBS01180
MM=2*ALI	PBS01190
DO 110 I=2,NST,MM	PBS01200
110 PTR2(I)=PTR2(I)-N	PBS01210
MM1=MM-1	PBS01220
DO 120 I=MM1,NST,MM	PBS01230
120 PTR2(I)=PTR2(I)+N	PBS01240
101 DO 95 I=1,NST	PBS01250
T2(I)=PTR1(I)	PBS01260
PTR1(I)=T1(I)	PBS01270
95 T1(I)=T2(I)	PBS01280
C	PBS01290
C STEADY STATE DISTRIBUTION DATA	PBS01300
C PTR1=STATE TO STATE TRANSITION VECTOR.	PBS01310
C PTR2=PROBABILITIES VECTOR FOR STATE TO STATE TRANSITIONS.	PBS01320
C	PBS01330
WRITE(8,1)(PTR1(I),I=1,NST)	PBS01340
WRITE(8,1)(PTR2(I),I=1,NST)	PBS01350
1 FORMAT(1X,2015)	PBS01360
K=1	PBS01370
DO 130 I=1,NST,2	PBS01380
T2(I)=K	PBS01390
T2(I+1)=K+N	PBS01400
KK=MOD((I+1)/2,M*ALT)	PBS01410
IF(KK.EQ.0) K=K+1	PBS01420
130 CONTINUE	PBS01430
C	PBS01440
C MEAN TRANSIENT RESPONSE DATA	PBS01450
C T1=STATE TO STATE TRANSITION VECTOR	PBS01460
C T2=PROBABILITY VECTOR FOR STATE TO STATE TRANSITIONS	PBS01470
C	PBS01480
WRITE(9,1)(T1(I),I=1,NST)	PBS01490
WRITE(9,1)(T2(I),I=1,NST)	PBS01500
STOP	PBS01510
END	PBS01520

```

C*****PBD00010
C                                     PBD00020
C      DETERMINATION OF LOOP PHASE PDF AND STEADY-STATE ERROR      PBD00030
C      FROM MARKOV CHAIN MODEL.                                     PBD00040
C                                                                     PBD00050
C      INPUT (READ ON DEVICE 8)                                     PBD00060
C      DATA FORMATS ARE SAME AS OUTPUT OF PHSTGEN                PBD00070
C      M=VALUE OF DIVIDE-BY M COUNTER                              PBD00080
C      N=VALUE OF DIVIDE-BY N COUNTER                              PBD00090
C      L=VALUE OF DIVIDE-BY L COUNTER                              PBD00100
C      POINT1=VECTOR CONTAINING POSSIBLE STATE TO STATE          PBD00110
C      TRANSITIONS.                                                PBD00120
C      POINT2=VECTOR CONTAINING THE POINTER FOR THE               PBD00130
C      PROBABILITIES OF THE STATE TO STATE                        PBD00140
C      TRANSITIONS.                                                PBD00150
C                                                                     PBD00160
C*****PBD00170
C      IMPLICIT REAL*8(A-H,O-Z)                                     PBD00180
C      INTEGER POINT1(7168),POINT2(7168),ALT                      PBD00190
C      DIMENSION POLD(3584),PNEW(3584),AO(64),A1(64),A(128),THETA(64) PBD00200
C      DIMENSION PHASE(64),PTEMP(3584)                             PBD00210
C                                                                     PBD00220
C      SET UPLL PARAMETERS                                         PBD00230
C                                                                     PBD00240
C      READ(8,1) N,M,L                                             PBD00250
C      7 FORMAT(3X,13,7X,13,9X,13)                                PBD00260
C      ALT=2*1.*1                                                  PBD00270
C      SNR=-40.                                                     PBD00280
C      MN=M*N                                                       PBD00290
C      IST=ALT*M*N                                                    PBD00300
C      NST=2*IST                                                    PBD00310
C      DO 20 I=1,IST                                                PBD00320
C      20 POLD(I)=1./IST                                             PBD00330
C                                                                     PBD00340
C      READ STATE TRANSFER VECTORS                                  PBD00350
C                                                                     PBD00360
C      READ(8,1) (POINT1(I),I=1,NST)                               PBD00370
C      READ(8,1) (POINT2(I),I=1,NST)                               PBD00380
C      1 FORMAT(1X,2015)                                           PBD00390
C                                                                     PBD00400
C      CALCULATE STATE TRANSFER PROBABILITIES                     PBD00410
C                                                                     PBD00420
C      DO 500 MM=1,13                                              PBD00430
C      PI=3.14159265                                                PBD00440
C      AC=DSQRT(2.00)*10.**((SNR/20.))                             PBD00450
C      PSI=-PI*PI/N                                                  PBD00460
C      DO 10 I=1,N                                                  PBD00470
C      RMEAN=AC*DSIN(PSI)/DSQRT(2.00)                               PBD00480
C      A1(I)=0.5+0.5*UERF(RMEAN)                                    PBD00490
C      AO(I)=1.-A1(I)                                               PBD00500
C      THETA(I)=PSI                                                  PBD00510
C      10 PSI=PSI+2.*PI/N                                           PBD00520
C      DO 12 I=1,N                                                  PBD00530
C      A(I)=AO(I)                                                    PBD00540
C      A(I+N)=A1(I)                                                 PBD00550

```

```

12 CONTINUE
JJ=1
C
C   CALCULATE STEADY-STATE LOOP STATE PROBABILITIES
C   BY ITERATION.
C
30 DIFF=0.0
JJ=JJ+1
K=1
J=1
DO 40 I=1,ISI
PNEW(I)=A(POINT2(J))*POLD(POINT1(J))+A(POINT2(J+1))*
(POLD(POINT1(J+1)))
IF(PNEW(I).LT.1.0-20) PNEW(I)=0.000
41 J=J+2
40 CONTINUE
SUM=0.
DO 50 I=1,ISI
50 SUM=SUM+PNEW(I)
DO 60 I=1,ISI
60 PNEW(I)=PNEW(I)/SUM
DO 61 I=1,ISI
61 PIEMP(I)=POLD(I)
DO 70 I=1,ISI
CNG=0.005*(PNEW(I)-POLD(I))
IF(CNG.GT.DIFF) DIFF=CNG
70 POLD(I)=PNEW(I)
IF(JJ.EQ.10000) GO TO 80
IF(DIFF.GT.1.E-8) GO TO 30
80 SUM=0.
C
C   CALCULATE STEADY-STATE LOOP PHASE PROBABILITIES
C   AND STANDARD DEVIATION OF LOOP PHASE.
C
C
STD=0.
PSI=-PI+PI/N
K=1
DO 90 I=1,ISI
SUM=SUM+POLD(I)
IF(MOD(I,M*ALI).NE.0) GO TO 90
PHASE(K)=SUM
K=K+1
STD=STD+SUM*PSI**2
SUM=0.
PSI=PSI+2.*PI/N
90 CONTINUE
STD=DSQRT(STD)
C
C   PROGRAM OUTPUT
C
91 WRITE(6,2)
2 FORMAT('1','DPLL PARAMETERS')
WRITE(6,8) SNR,N,M,ALI
8 FORMAT(1X,'SNR=',F7.1,5X,'N=',13,5X,'M=',13,5X,'ALI=',13,7)
WRITE(6,3) STD,JJ,DIFF

```

```

PBD00560
PBD00570
PBD00580
PBD00590
PBD00600
PBD00610
PBD00620
PBD00630
PBD00640
PBD00650
PBD00660
PBD00670
PBD00680
PBD00690
PBD00700
PBD00710
PBD00720
PBD00730
PBD00740
PBD00750
PBD00760
PBD00770
PBD00780
PBD00790
PBD00800
PBD00810
PBD00820
PBD00830
PBD00840
PBD00850
PBD00860
PBD00870
PBD00880
PBD00890
PBD00900
PBD00910
PBD00920
PBD00930
PBD00940
PBD00950
PBD00960
PBD00970
PBD00980
PBD00990
PBD01000
PBD01010
PBD01020
PBD01030
PBD01040
PBD01050
PBD01060
PBD01070
PBD01080
PBD01090
PBD01100

```

FILE: PBDPL2 FORTRAN A

OHIO UNIVERSITY AVIONICS ENGINEERING CENTER

```
3  FORMAT(1X,'STD DEV=',F8.3,5X,'NO. OF ITER=',I6,5X,'DIFF=',E12.4,/)PBD01110
   WRITE(6,4)(THETA(I),PHASE(I),I=1,N)PBD01120
4  FORMAT(1X,'PHASE DIF=',E12.4,5X,'PDF=',E12.4)PBD01130
500 SNR=SNR+5PBD01140
100 STOPPBD01150
   ENDPBD01160
```

```

C*****PBD00010
C      DETERMINATION OF MEAN TIME TO LOCK FOR INITIAL PHASE OFFSET PBD00020
C      FROM MARKOV CHAIN MODEL. PBD00030
C PBD00040
C      INPUT (READ ON DEVICE 8) PBD00050
C      DATA FORMAT SAME AS OUTPUT OF PBSTGEN PBD00060
C      M=VALUE OF DIVIDE-BY M COUNTER PBD00070
C      N=VALUE OF DIVIDE-BY N COUNTER PBD00080
C      L=VALUE OF DIVIDE-BY L COUNTER PBD00090
C      POINT1=VECTOR CONTAINING THE POSSIBLE STATE TO PBD00100
C      STATE TRANSITIONS. PBD00110
C      POINT2=VECTOR CONTAINING THE POINTER FOR THE STATE PBD00120
C      TO STATE TRANSITION PROBABILITIES. PBD00130
C PBD00140
C PBD00150
C*****PBD00160
C      IMPLICIT REAL*8(A-H,O-Z) PBD00170
C      INTEGER POINT1(7168),POINT2(7168),ALT PBD00180
C      DIMENSION POLD(3584),PNEW(3584),AO(64),A1(64),A(128),THETA(64) PBD00190
C      DIMENSION PHASE(64),PTEMP(3584) PBD00200
C PBD00210
C      SET DPLL PARAMETERS PBD00220
C PBD00230
C      READ(8,/) N,M,L PBD00240
C      7 FORMAT(3X,13,7X,13,9X,13) PBD00250
C      ALT=2*L+1 PBD00260
C      SNR=-40. PBD00270
C      MN=M*N PBD00280
C      IST=ALT*M*N PBD00290
C      NST=2*IST PBD00300
C      DO 20 I=1,IST PBD00310
C      20 POLD(I)=1. PBD00320
C      JL=IST/2-(ALT/2) PBD00330
C      JLI=JL-(M-1)*ALT PBD00340
C      JLU=JLI+((2*M)-1)*ALT PBD00350
C PBD00360
C      READ STATE TRANSFER VECTORS PBD00370
C PBD00380
C      READ(8,1)(POINT1(I),I=1,NST) PBD00390
C      READ(8,1)(POINT2(I),I=1,NST) PBD00400
C      1 FORMAT(1X,20I5) PBD00410
C PBD00420
C      CALCULATE STATE TRANSFER PROBABILITIES PBD00430
C PBD00440
C      DO 500 MM=1,13 PBD00450
C      PI=3.14159265 PBD00460
C      AC=DSQRT(2.00)*10.**((SNR/20.)) PBD00470
C      PSI=-PI+PI/N PBD00480
C      DO 10 I=1,N PBD00490
C      RMEAN=AC*DSIN(PSI)/DSQRT(2.00) PBD00500
C      A1(I)=0.5+0.5*DERF(RMEAN) PBD00510
C      AO(I)=1.-A1(I) PBD00520
C      THETA(I)=PSI PBD00530
C      10 PSI=PSI+2.*PI/N PBD00540
C      DO 12 I=1,N PBD00550

```

```

      A(I)=A0(I)
      A(I+N)=A(I)
12  CONTINUE
      JJ=1
C
C      CALCULATE MEAN TIME TO LOCK
C
30  DIFF=0.0
      JJ=JJ+1
      K=1
      J=1
      DO 40 I=1,ISI
        PNEW(I)=A(PGINT2(J))*POLD(PGINT1(J))+A(PGINT2(J+1))*
        POLD(PGINT1(J+1))+1.
        IF(PNEW(I).LT.1.D-20) PNEW(I)=0.000
41  J=J+2
40  CONTINUE
      DO 71 JJ1=JL1,JLU,ALT
61  PNEW(JJ1)=0.
      DO 70 I=1,ISI
        CNG=0ABS(PNEW(I)-POLD(I))
        IF(CNG.GT.DIFF) DIFF=CNG
70  POLD(I)=PNEW(I)
        IF(JJ.EQ.10000) GO TO 80
        IF(DIFF.GT.1.E-5) GO TO 30
80  SUM=0.
      K1=ALT/2+1
      K2=M*ALT
      J=1
      DO 110 I=K1,ISI,K2
        POLD(J)=PNEW(I)
110  J=J+1
C
C      PROGRAM OUTPUT
C
91  WRITE(6,2)
2  FORMAT('1','DPLL PARAMETERS')
      WRITE(6,8) SNR,N,M,ALT
8  FORMAT(1X,'SNR=',F7.1,5X,'N=',I3,5X,'M=',I3,5X,'ALT=',I3,/)
      WRITE(6,3) JJ,DIFF
3  FORMAT(1X,'NO. OF ITER=',I6,5X,'DIFF=',E12.4,/)
      WRITE(6,4) (THETA(I),POLD(I),I=1,N)
4  FORMAT(1X,'PHASE DIFF=',E12.4,5X,'MEAN NO. SAMP.',E12.4)
500 SNR=SNR+5
100 STOP
      END

```

PBD00560
 PBD00570
 PBD00580
 PBD00590
 PBD00600
 PBD00610
 PBD00620
 PBD00630
 PBD00640
 PBD00650
 PBD00660
 PBD00670
 PBD00680
 PBD00690
 PBD00700
 PBD00710
 PBD00720
 PBD00730
 PBD00740
 PBD00750
 PBD00760
 PBD00770
 PBD00780
 PBD00790
 PBD00800
 PBD00810
 PBD00820
 PBD00830
 PBD00840
 PBD00850
 PBD00860
 PBD00870
 PBD00880
 PBD00890
 PBD00900
 PBD00910
 PBD00920
 PBD00930
 PBD00940
 PBD00950
 PBD00960
 PBD00970
 PBD00980
 PBD00990
 PBD01000
 PBD01010

C. Binary Phase-Locked Loop Design. The following sections discuss in detail the design of the hardware DPLL used to validate the Markov chain model. The block diagram for the DPLL may be found in Figure 6-1.

a. Binary Phase Detector. Referring to the schematic of Figure C-1, the binary phase detector operates by sampling the binary input signal f_c and producing a complemented pulse output on either the count-up or count-down line. This function is produced by three D-type flip-flops as follows. The OPEN SW signal is applied to the clock input of ff B3 while the binary signal f_c is applied to the data input of the same ff. Thus on a positive transition of OPEN SW the value of f_c is latched, giving the sampled value IN at B3's Q output. The two ff's of A6 are initially set to the ONE state so that when SET LAT latches the input values to the ff's (IN and $\overline{\text{IN}}$), one ff goes to the ZERO state while the other remains in the ONE state. The SET LAT signal is followed by the $\overline{\text{CLR LAT}}$ signal which sets both ff's to the ONE state. Thus the count-up or count-down are produced in a mutually exclusive manner when one of the ff's of A6 toggles HIGH LOW HIGH while the other remains HIGH.

b. DIVIDE-by L Saturating Counter. The saturating up-down divide-by L counter is shown in Figure C-2. The counting function is performed by two serially connected 74193 4-bit synchronous up-down counters. The output states of the counter is detected by a logic network to produce the $\overline{\text{INHIB UP}}$ and $\overline{\text{INHIB DN}}$ signals that will inhibit the UP and DN clock signals respectively. For example, if switches SW1 through SW6 are closed, then for a counter state of 0000 0001 the $\overline{\text{INHIB UP}}$ signal will be TRUE, thus inhibiting the UP clock signal and saturating the counter at that value. Note, a DN clock signal will still count the counter

to the 0000 0000 state. Similarly, for all switches closed and the counter in state 1111 1111 the $\overline{\text{INHIB DN}}$ signal will be TRUE and the counter is saturated at that state. If switch SW1 is opened while all other switches remain closed, then the counter saturation states will be 0000 0011 and 1111 1101 (± 3). Continuing in this manner, the counter saturation states may be selected to be $\pm 2^i - 1$, $i = 1, 2, \dots, 7$ by opening switches SW1 through SW ($i-1$) while all other switches remain closed.

c. Divide-by ($M \times N$) Counter. The schematic for the divide by ($M \times N$) counter along with the buffer register and adder are shown in Figure C-3. The counter consists of three serially connected 74193 4-bit up-down counters. Following an UP or DN clock signal the value of the divide-by ($M \times N$) counter is loaded into the 12-bit buffer register composed of two 74174 hex D-type flip-flops by the LOAD2 signal. The values of the 12-bit buffer and the divide-by L saturating counter compose the inputs to the adder circuit consisting of three 7483 4-bit binary adders. The output of the 12-bit adder is applied to the preset terminals of the divide-by ($M \times N$) counter. If the type select switch is set for first-order operation, the adder output will not be loaded into the divide-by ($M \times N$) counter. If a second-order loop is selected, the LOAD1 signal will be applied to the load inputs of the 74193's thus presetting the divide-by ($M \times N$) counter to the value of the 12-bit adder output.

d. Variable Phase Reference Clock. Referring to Figure C-4, six selectable contiguous bits from the divide-by ($M \times N$) counter are applied to one input side of a binary magnitude comparator formed by two 7485 4-bit magnitude comparators. The other input to the comparator is obtained from the lower N bits of the two series-connected 74193 counters that are being clocked at a rate of 2^N

time the input frequency. Thus the EQUALS output pin 6 of A31, of the magnitude comparator will be a ONE when the two inputs are equal and this will occur at a rate equal the input frequency f_c . The circuit formed by A33 and A34 prevents the sample output from occurring at a rate greater than f_c . For example, if pin 8 of A34 is HIGH then when the magnitude comparator detects the equal condition, pin 5 of A34 is set HIGH causing the phase detector to sample the input signal. The SAMPLE signal is reset LOW by the LOAD1 signal at the end of the loop phase update. However, since the $(M + N)$ -bit counter may have counted up during the last phase update and the counters of the reference clock have also counted up, it is possible for the magnitude comparator to detect equal states occurring at a rate of $2^N * f_c$. Thus it is necessary for the $\overline{\text{LOAD1}}$ signal to reset pin 8 of A33 to a ZERO so that another sample cannot occur until the reference clock counter loads a ZERO into that flip-flop.

e. Control Logic. A timing diagram for the necessary control waveforms and the logic schematic used in their generation are given in Figures C-5 and C-6, respectively. Use of the control waveforms have been described in the previous sections.

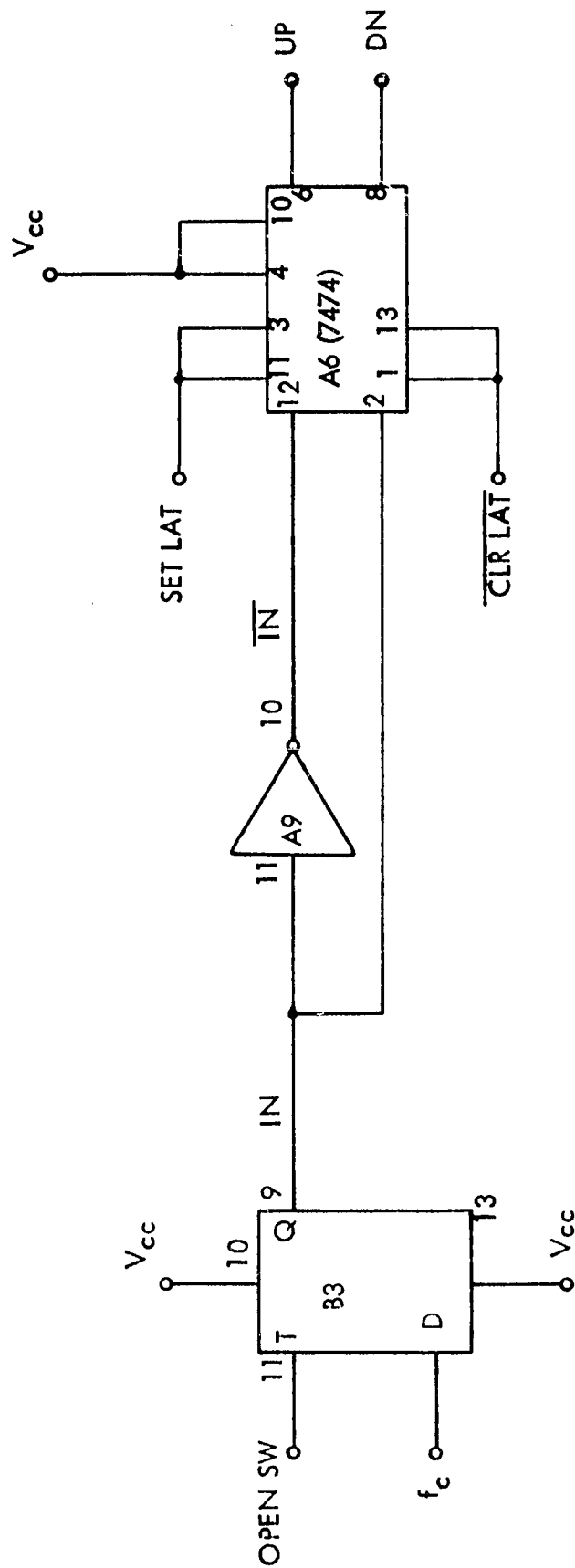


Figure C-1. DPLL Binary Phase Detector.

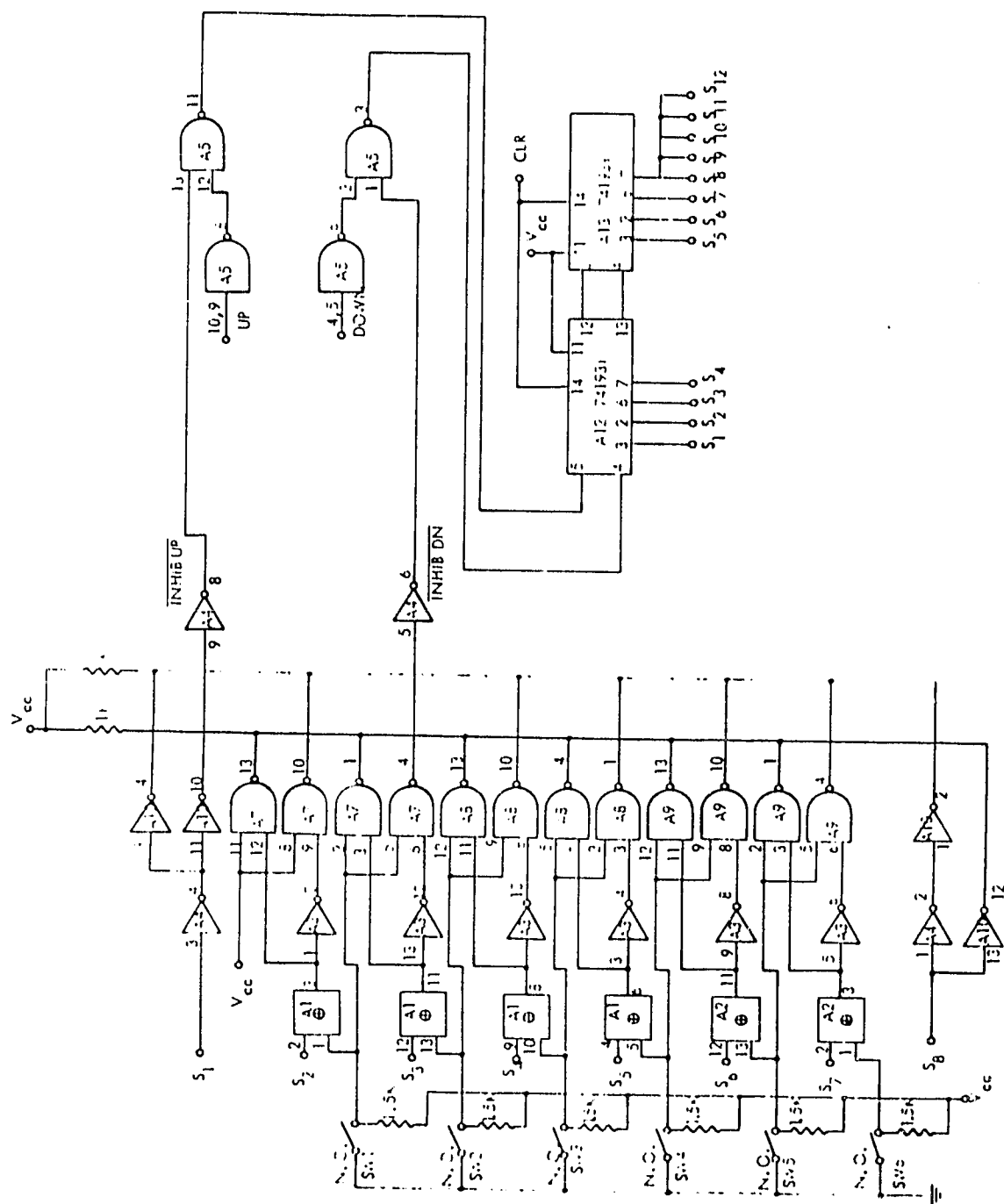


Figure C-2. S-8 Bit Saturating Counter.

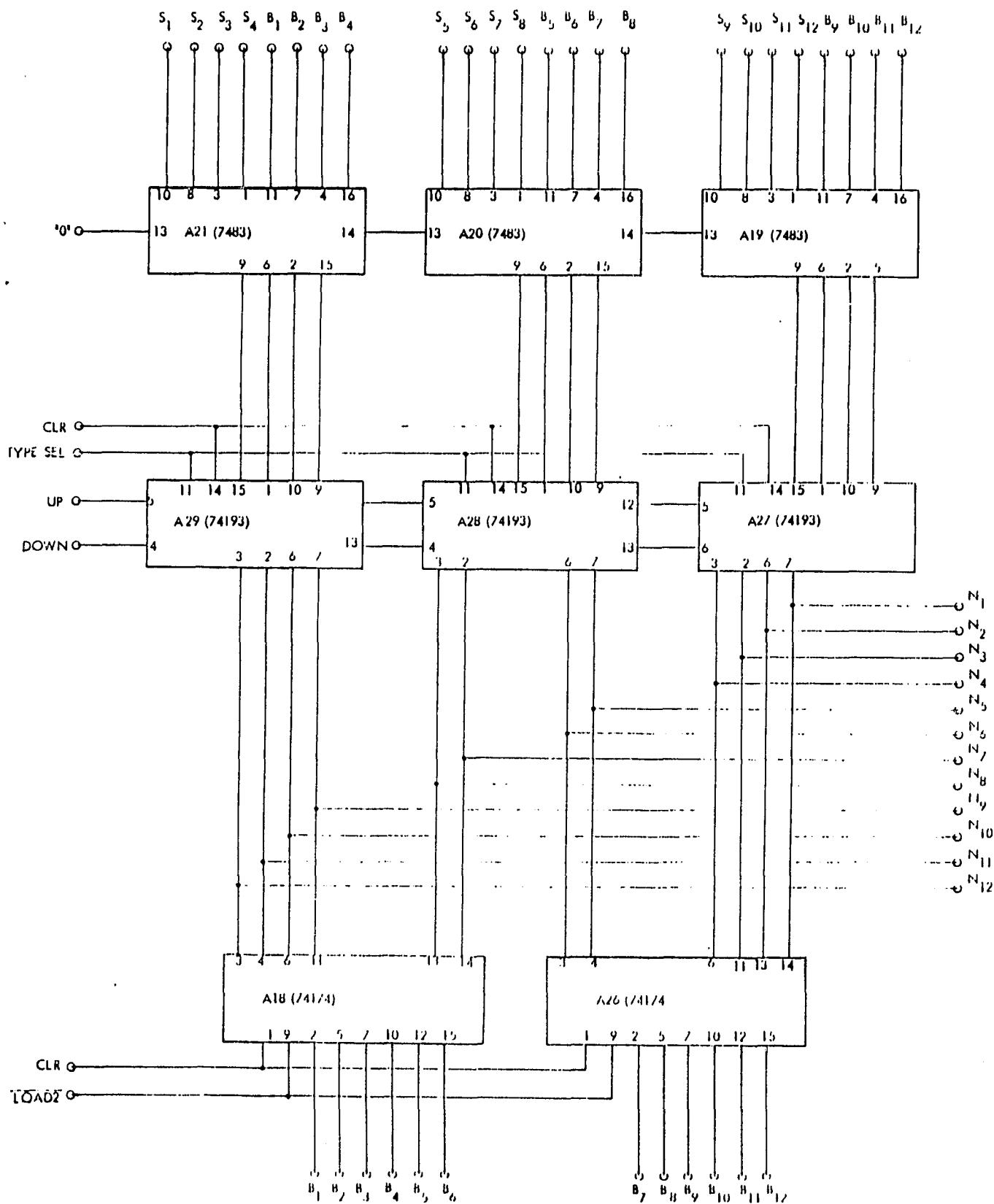
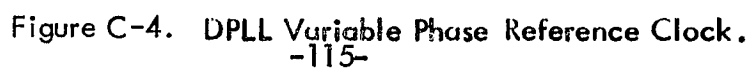


Figure C-3. (M-N) Bit Counter with Adder Function.



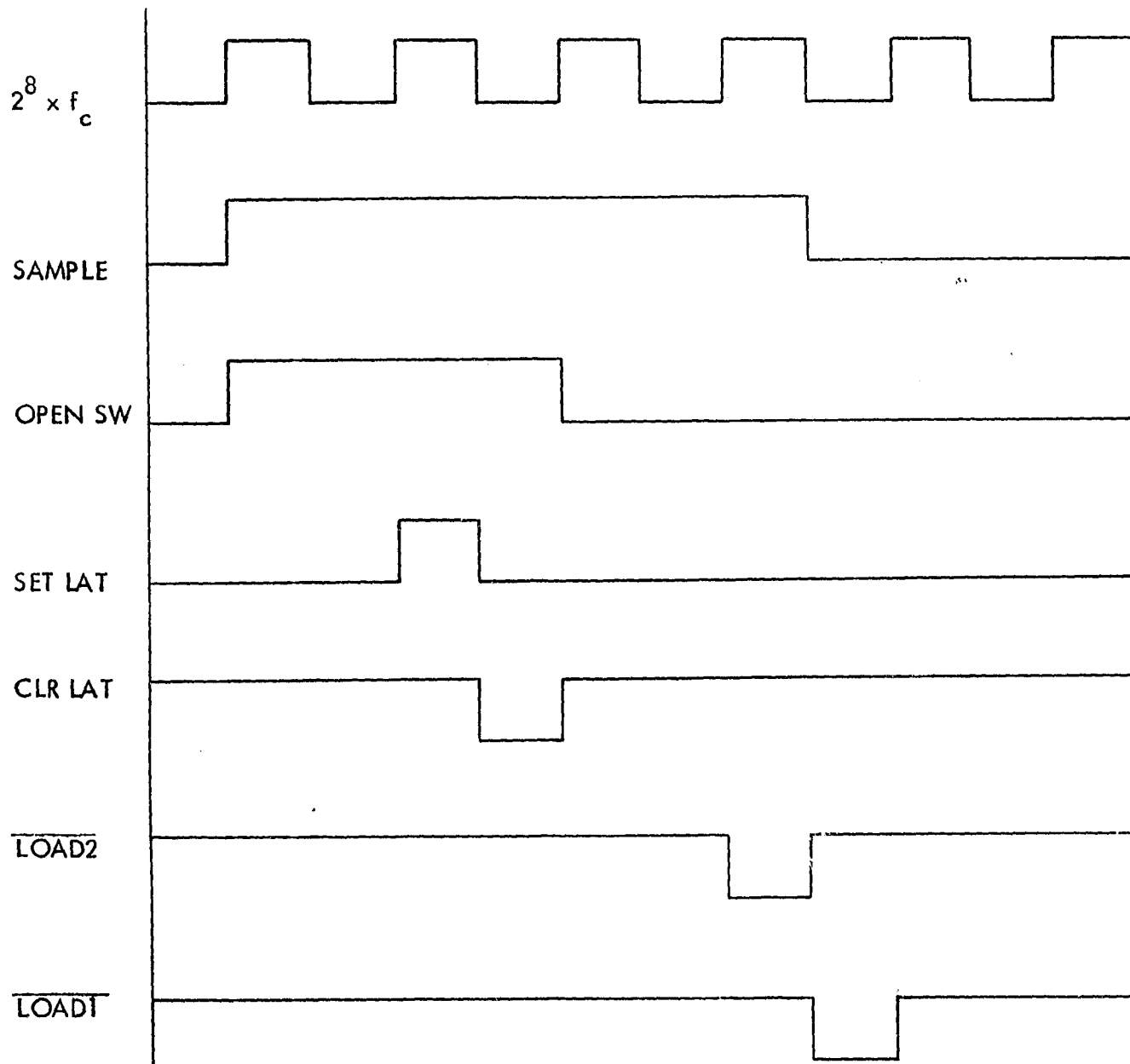


Figure C-5. Control Waveforms.

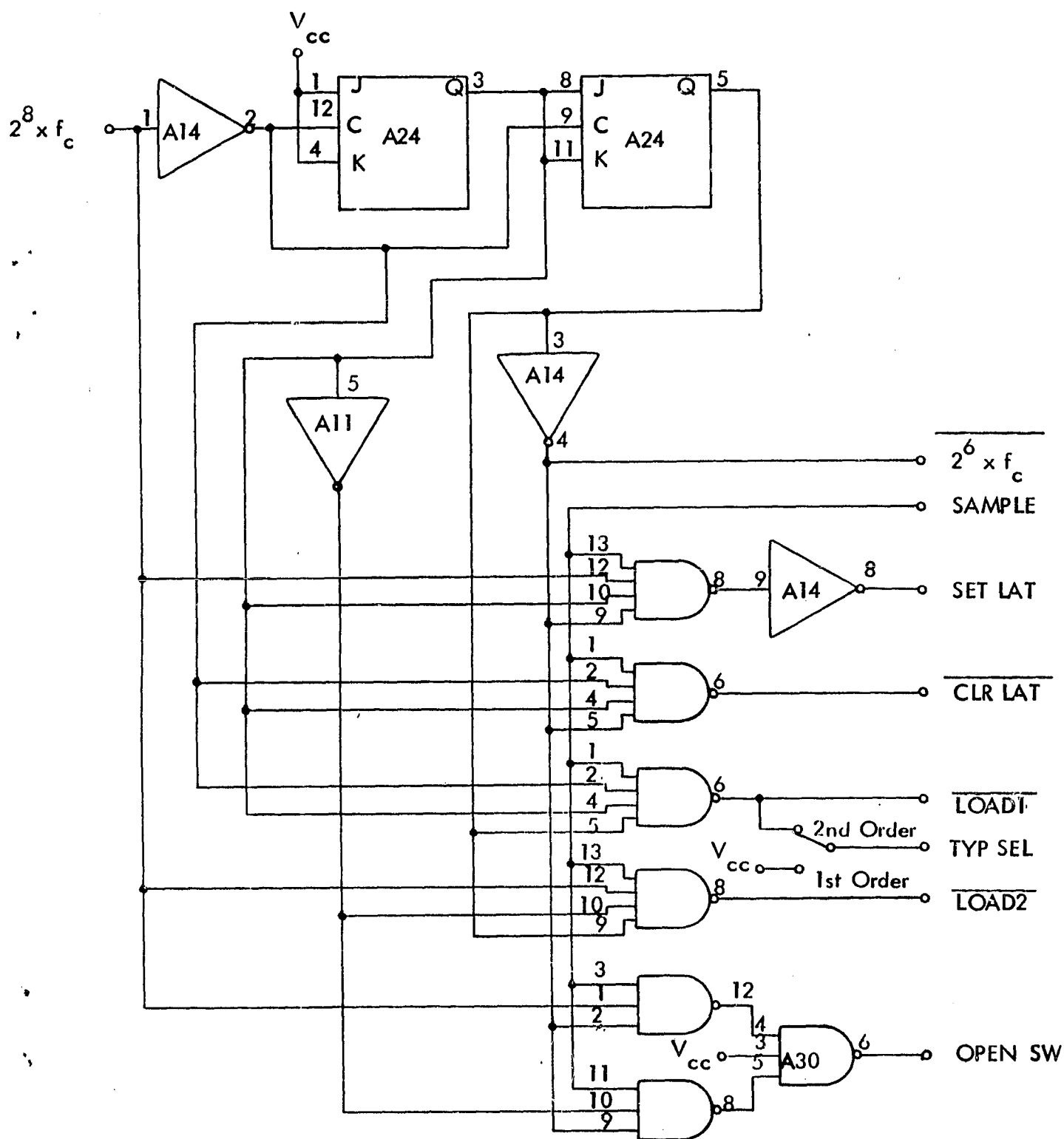


Figure C-6. Control Logic.

D. Digital Low-Pass Filter. Consider a divide-by M binary up/down counter whose input $x(t) = \pm 1$ is clocked at a rate $f_c = 1/T$. The contents of the counter will be the sum of the previous $M-1$ inputs plus the present input, so that the value of the counter $y(t)$ may be expressed by the discrete equation

$$y(kT) = \frac{1}{M} \sum_{i=0}^{M-1} x[(k-i)T] \quad (D-1)$$

up to the point of counter overflow or underflow. Taking the Fourier transform of (D-1) gives

$$Y(j\omega) = \frac{1}{M} \sum_{i=0}^{M-1} X(j\omega) e^{-j i \omega T} \quad (D-2)$$

which gives the transfer function for the counter as,

$$H(j\omega) = \frac{1}{M} \sum_{i=0}^{M-1} e^{-j i \omega T} \quad (D-3a)$$

$$= \frac{1}{M} \frac{1 - e^{-j \omega T M}}{1 - e^{-j \omega T}} \quad (D-3b)$$

$$= \frac{1}{M} \frac{\sin\left(\frac{\omega T M}{2}\right)}{\sin\left(\frac{\omega T}{2}\right)} e^{-j \frac{\omega T}{2} (M-1)} \quad (D-3c)$$

Substituting for $T = 2\pi/\omega_c$ in (D-3c) gives the magnitude-squared function for the counter as,

$$\left| H(j\omega) \right|^2 = \left[\frac{1}{M} \frac{\sin(\pi M \omega/\omega_c)}{\sin(\pi \omega/\omega_c)} \right]^2 \quad (D-4)$$

which has been plotted in Figure D-1 for $M = 2, 4$, and 8 . As can be seen from Figure D-1, the binary up/down counter acts as a digital low-pass filter whose bandwidth decreases as M increases.

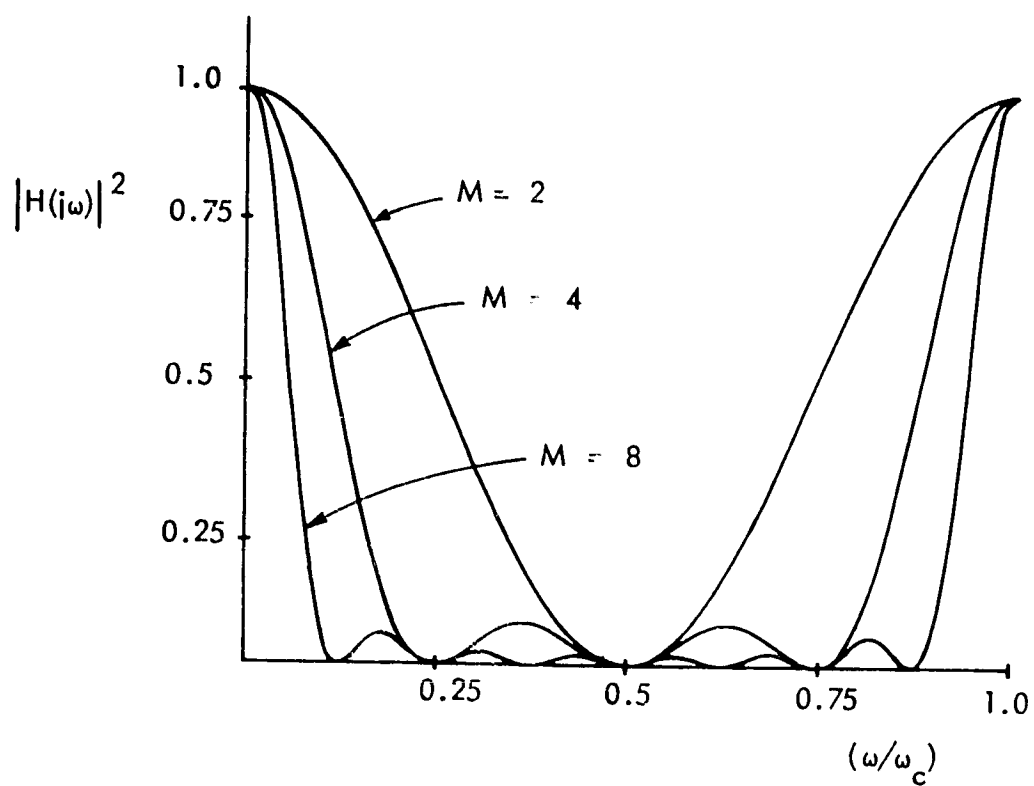


Figure D-1. Magnitude-Squared for Digital Low-Pass Filter.